Distributed Sub-gradient Method for Delay Tolerant Networks

Riccardo Masiero, Giovanni Neglia

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Distributed Sub-gradient Method for Delay Tolerant Networks

Riccardo Masiero — Giovanni Neglia

N° 7345

June 2010

Networks and Telecommunications

ISSN 0249-6399 ISRN INRIA/RR--7345--FR+ENG

INSTITUT NATIONAL
DE RECHERCHE
EN INFORMATIQUE
ET EN AUTOMATIQUE

INRIA

centre de recherche
SOPHIA ANTIPOLIS - MÉDITERRANÉE

Distributed Sub-gradient Method for Delay Tolerant Networks

Riccardo Masiero*†, Giovanni Neglia†

Theme : Networks and Telecommunications
Networks, Systems and Services, Distributed Computing
Équipe-Projet MAESTRO

Rapport de recherche n° 7345 — June 2010 — 38 pages

Abstract: In this report we consider that nodes in a Delay Tolerant Network (DTN) may collaborate to minimize the sum of local objective functions, depending in general on some parameters or actions of all the nodes in the network. If the local objective functions are convex, it can be adopted a recently proposed computation framework that relies on local sub-gradient methods and consensus algorithms to average each node information. Existing convergence results for this framework can be applied to DTNs only in the case of synchronous node operation and simple mobility models without memory. We address both these issues. First, we prove convergence to the optimal solution for a more general class of mobility processes. Second, we show that, under asynchronous operations, a straight application of the original method would lead to sub-optimal solutions and we propose some changes to solve this problem. As a particular case study, we show how the framework can be applied to optimize the dissemination of dynamic content in a DTN.

Key-words: delay tolerant networks, distributed optimization, consensus, sub-gradient method

* Department of Information Engineering, University of Padova 6/B, Via G. Gradenigo, I–35131 Padova (PD), Italy. E-mail: riccardo.masiero@dei.unipd.it.

† INRIA-MAESTRO, 2004 route des Lucioles, BP 93, FR-06902 Sophia Antipolis (BIOT), France. E-mails: {riccardo.masiero, giovanni.neglia}@sophia.inria.fr.

Centre de recherche INRIA Sophia Antipolis – Méditerranée
2004, route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex
Téléphone : +33 4 92 38 77 77 — Télécopie : +33 4 92 38 77 65

Méthode de sous-gradient distribuée dans les réseaux tolérant les délais

Résumé : Dans ce rapport, nous considérons que les nœuds dans un réseau tolérant les delais (Delay Tolerant Network, DTN) peuvent collaborer afin de minimiser la somme des fonctions objective locales, qui dépendent en général des paramètres ou des actions de tous les nœuds du réseau. Si les fonctions objectif locales sont convexes, il peut être adopté une méthodologie récemment proposé, qui s'appuie sur le calcul du sous-gradient de la fonction locale et les algorithmes de consensus pour faire la moyenne de l'information de chaque nœud. Les résultats de convergence existants pour cette méthodologie peuvent être appliqués aux DTNs uniquement dans le cas de operation synchrone des nœuds et pour des modèles de mobilité simples, sans mémoire. Nous abordons ces deux questions. Tout d'abord, nous prouvons la convergence de la méthode à la solution optimale pour une classe plus générale des processus de mobilité. Deuxièmement, nous montrons que, dans le cas de opération asynchrone, une application directe de la méthode originale conduit à des solutions sous-optimales et nous proposons quelques modifications pour résoudre ce problème. Comme étude de cas particulier, nous montrons comment le cadre peut être utilisé pour optimiser la diffusion de contenus dynamiques dans un DTN.

Mots-clés : réseau tolérant les délais, optimisation distribuée, consensus, méthode de sous-gradient

1 Introduction

In networking applications, the performance of a Delay Tolerant Network (DTN) is a global measure that depends on decisions (i.e., protocol rules) and variables (i.e., protocol parameters) at each network node. Hence, the optimization of any given network protocol can be described as a global optimization problem which is governed by the local actions taken by each node. As an example, the message delivery delay and the energy consumption under the gossip protocol [1, 2] depend on the message forwarding probabilities which can be locally and independently calculated by each node. To further complicate matters, local (but globally optimal) decisions at different nodes are not independent and the optimal configuration is in general heterogeneous and depends on the specific scenario, as different nodes have different roles in the network. Given this, it may be not possible to compute optimal protocol rules and parameters off-line prior to network deployment. In addition, the disconnected nature of DTNs calls for on-line and distributed approaches to optimization where, in practice, each node has access to local variables and rules which can only be set according to what occurs within its immediate surroundings (the visibility scope of the node).

The authors of [3] present a distributed solution to this problem for the case where the global optimization target f can be expressed as sum of M convex functions f_i and each node i only knows the corresponding function f_i, referred to as the *local objective function*. Many performance metrics of interest have this decomposition property. For example, this is the case of performance metrics related to nodes (e.g., energy consumption at each node) or to messages (e.g., delivery time, delivery probability, number of copies in the network). In either case, the metrics can naturally be expressed as a sum of local cost functions relative to each node. Convexity may be not guaranteed, but when this assumption does not hold the system converges in general to a sub-optimal but still desirable solution.

In the framework proposed in [3], and later extended in [4], nodes optimize their own local objective functions through a *sub-gradient method*, where they try to reach agreement on their local estimates by occasionally exchanging their local information and averaging it, like in a consensus problem [5, 6]. Within this approach, referred to as the *distributed sub-gradient method*, the local estimate of each node is proven to converge to the optimal solution under certain assumptions. However, two of these assumptions appear to be particularly restrictive for practical use in DTN scenarios. Specifically, the node mobility process should have strict deterministic bounds on the inter-meeting times between nodes, see [3], or it should be memory-less, see [4]. Neither of these conditions is in general satisfied in a real network. Second, all nodes should update their estimates at the same time, but synchronicity is difficult to achieve in such a disconnected scenario.

As a first contribution, in this report we relax these assumptions. First, we prove that the distributed sub-gradient method also converges under a more general Markovian mobility model with memory in the meeting process. In addition, we show that a direct application of the framework when nodes operate asynchronously may introduce a bias, leading to convergence to a sub-optimal solution. Hence, we propose some adjustments, and show by simulations that they are able to correct the bias.

Furthermore, inspired by the work in [7], we propose a possible application of the framework to a DTN scenario where a Service Provider (SP) disseminates a dynamic content over a mobile social network, with the help of the users that opportunistically share among themselves content updates. In this context the SP should decide how to allocate its bandwidth optimally, and to this purpose it needs to collect information about node utility functions and node meeting rates. We show that distributed sub-gradients can be effectively used to let the nodes perform such optimization.

The report is organized as follows. In Sec. 2 we review the distributed sub-gradient method, whilst in Sec. 3 we show how to apply it to DTNs and motivate our work. As original contributions, in Sec. 4 we extend the results in [3] and [4] to more general network mobility models and in Sec. 5, we study how the presented framework needs to be extended to cope with asynchronous node operations. Hence, in Sec. 6, we illustrate a possible DTN application exploiting the distributed sub-gradient method. Sec. 7 concludes the report.

2 Distributed Sub-gradient Method's Overview

In this Section we review the main results in [3, 4] on convergence and optimality of the distributed sub-gradient method when a random network scenario is considered.

Let us consider a set of M nodes (agents), that want to cooperatively solve the following optimization problem:

Problem 1 (Global Optimization Problem)**.** *Given M convex functions $f_i(\mathbf{x}) : \mathbb{R}^N \to \mathbb{R}$, determine:*

$$\mathbf{x}^* \in \operatorname*{argmin}_{\mathbf{x}\in\mathbb{R}^N} f(\mathbf{x}) = \sum_{i=1}^{M} f_i(\mathbf{x}) \ .$$

Clearly, for the above problem we assume that a feasible solution exists. The difficulty of the task arises from the fact that agent i, for $i = 1, 2, \cdots M$, only knows the corresponding function $f_i(\mathbf{x})$, namely its *local objective function*. For example f_i could be a performance metric relative to node i, and f could indicate global network performance.

If the functions f_i are differentiable, each node could apply a gradient method to its function f_i to generate a sequence of local estimates, but this would lead to M biased estimates of the solution of problem 1. In [3] and [4], it is shown that if nodes perform a gradient method but are also able to *average* their local estimates, under opportune conditions, these estimates all converge to a point of minimum of f, i.e., $\mathbf{x}^*$.

In particular, a time slotted system is assumed, where, at the end of a slot, each node i communicates its local estimate to a subset of all the other nodes, and then updates the estimate according to the following equation[1]:

$$\mathbf{x}^i(k+1) = \sum_{j=1}^{M} a_{ij}(k)\mathbf{x}^j(k) - \gamma(k)\mathbf{d}^i(k) \ , \tag{1}$$

where the vector $\mathbf{d}^i(k) \in \mathbb{R}^M$ is a sub-gradient[2] of agent i's objective function $f_i(\mathbf{x})$ computed at $\mathbf{x} = \mathbf{x}^i(k)$, the scalar $\gamma(k) > 0$ is the step-size of the sub-gradient algorithm at iteration k, and $a_{ij}(k)$ are non-negative weights, such that $a_{ij}(k) > 0$ if and only if node i has received node j's estimate at the step k and $\sum_{j=1}^{M} a_{ij}(k) = 1$. We denote by $\mathbf{A}(k)$ the matrix whose elements are the weights, i.e. $[\mathbf{A}(k)]_{ij} = a_{ij}(k)$.

We observe that the first addend in the right hand side of 1 corresponds to average according to a consensus algorithm [5].

[3] proves that the iterations (1) generate sequences converging to a minimum of f under the following set of conditions:

1. the step-size $\gamma(k)$ is such that $\sum_{i=1}^{\infty} \gamma(k) = \infty$ and $\sum_{i=1}^{\infty} \gamma(k)^2 < \infty$;
2. the gradient of each function f_i is bounded;
3. each matrix $\mathbf{A}(k)$ is symmetric (then doubly stochastic);

[1] In this report all the real valued vectors are assumed to be column vectors.

[2] $\mathbf{d}^i \in \mathbf{R}^N$ is a sub-gradient of the function f_i at $\mathbf{x}^i \in \mathrm{dom}(f_i)$ iif $f_i(\mathbf{x}^i) + (\mathbf{d}^i)^T(\mathbf{x} - \mathbf{x}^i) \leq f_i(\mathbf{x})$ for all $\mathbf{x} \in \mathrm{dom}(f_i)$.

4. it exists $\eta > 0$, such that $a_{ii}(k) > \eta$ and, if $a_{ij}(k) > 0$, then $a_{ij}(k) \geq \eta$;

5. the information of each agent i reaches every other agent j (directly or indirectly) infinitely often;

6′) there is a deterministic bound for the intercommunication interval between two nodes.

We better formalize conditions 5 and 6′ (resp. 6″). Consider the graph $(\mathcal{V}, E_\infty)$, where $\mathcal{V}$ is the set of nodes and the edge (i,j) belongs to E_∞ if nodes i and j communicate infinitely often (i.e. if $a_{ij}(k)$ is positive for infinite values k). Condition 5 imposes that the graph $(\mathcal{V}, E_\infty)$ is (strongly) connected. Condition 6′ requires that there is a positive integer constant B, such that two nodes communicating infinitely often, communicate at least once every B slots, i.e., if $(i,j) \in E_\infty$, then $\max\{a_{ij}(k), a_{ij}(k+1), \cdots, a_{ij}(k+B-1)\} > 0$.

In [4], the inter-meeting times are not deterministically bounded, but matrices are required to be independently and identically distributed. In fact, condition 6′ is replaced by the following one:

6″) matrices $\mathbf{A}(k)$ are i.i.d. random matrices.

In such case, $(i,j) \in E_\infty$ if and only if $\mathrm{E}[a_{ij}(k) > 0]$. Note that, when the matrices $\mathbf{A}(k)$ are random (like in 6″), condition 5 requires the matrix $\mathrm{E}[\mathbf{A}(k)]$ to be irreducible and aperiodic.

Both papers address also the case when the gradient step-size does not vanish, but it is kept constant ($\gamma(k) = \gamma$). In this case, the sequence of estimates $\mathbf{x}^i$ does not converge in general to a point of minimum of f, but it may keep oscillating around one of such point. It is possible to bound the difference between the values that f assumes at the points of a smoothed average of $\mathbf{x}^i$ and the minimum of f. As it is intuitive, the smaller γ, the smaller such difference.

We are going to provide an intuitive explanation of why the results on convergence and optimality hold, and an outline of the proofs in [3, 4]. This will be useful for our following extensions. We first formulate (1) in matrix form as follows:

$$\mathbf{X}(k+1) = \mathbf{A}(k)\mathbf{X}(k) - \gamma(k)\mathbf{D}(k) \ , \tag{2}$$

where

$$\mathbf{X}(k) \stackrel{\text{def}}{=} \left[\mathbf{x}^1(k+1), \cdots, \mathbf{x}^i(k+1), \cdots, \mathbf{x}^M(k+1)\right]^T$$

and

$$\mathbf{D}(k) \stackrel{\text{def}}{=} \left[\mathbf{d}^1(k+1), \cdots, \mathbf{d}^i(k+1), \cdots, \mathbf{d}^M(k+1)\right]^T \ .$$

This equation iteratively leads to (see Appendix A)

$$\begin{aligned} \mathbf{X}(k+1) \quad = \quad & \mathbf{A}_{(1)}^{(k)}\mathbf{X}(1) - \sum_{s=2}^{k} \mathbf{A}_{(s)}^{(k)}\gamma(s-1)\mathbf{D}(s-1) + \\ & -\gamma(k)\mathbf{D}(k) \ , \end{aligned} \tag{3}$$

where $\mathbf{A}_{(s)}^{(k)}$, with $s, k \geq 1$ and $s \leq k$, is the *backward matrix product*, i.e., $\mathbf{A}_{(s)}^{(k)} = \mathbf{A}(k)\mathbf{A}(k-1)\cdots\mathbf{A}(s)$. We introduce the average of all the nodes estimates, $\mathbf{y}(k) \in \mathbb{R}^M$, defined as:

$$\mathbf{y}(k)^T = \frac{1}{M}\mathbf{1}^T\mathbf{X}(k) \ .$$

By (2), we obtain

$$\begin{aligned}\mathbf{y}(k+1)^T &= \frac{1}{M}\mathbf{1}^T\mathbf{A}(k)\mathbf{X}(k) - \mathbf{1}^T\frac{\gamma(k)}{M}\mathbf{D}(k) = \\ &= \mathbf{y}(k)^T - \frac{\gamma(k)}{M}\mathbf{1}^T\mathbf{D}(k)\ . \end{aligned} \tag{4}$$

Assume for a moment that $\mathbf{x}^i(k) = \mathbf{x}^j(k) = \mathbf{y}(k)$, for each i and j, then sub-gradients $\mathbf{d}^i(k)$ are all evaluated in $\mathbf{y}(k)$ and $\mathbf{1}^T\mathbf{D}(k)$ is a sub-gradient of the global function f evaluated in $\mathbf{y}(k)$. Thus, the above equation (4) corresponds to a basic iteration of a sub-gradient method for the global function f. The intuitive explanation of the result is that averaging keeps the estimates of $\mathbf{x}^i(k)$, for all i, close each other (and then close to $\mathbf{y}(k)$) and makes the local sub-gradients updates equivalent to a sub-gradient update for the global function f.

We illustrate this through a simple toy example that we are going to use different times across this report. Consider three nodes, labeled as 1, 2 and 3. Their local objective functions are $f_1(x) = f_2(x) = x(x-1)/2$ and $f_3(x) = 2x^2$, where $x \in \mathbb{R}$. Then the global function is $f(x) = \sum_i f_i(x) = 3x^2 - x$, it has minimum value equal to $-1/12$ and a unique point of minimum in $x = 1/6$. The weight matrices $\mathbf{A}(k)$ are i.i.d. random matrices. At each step $\mathbf{A}(k)$ is equal to one of the following three matrices

$$\begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 1 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix}, \tag{5}$$

with probability 2/3, 1/6 and 1/6, respectively. Fig. 1 shows the evolution of the estimates at the 3 nodes, when the algorithm is applied with $\gamma(k) = 1/k$. We can see that state's estimates tends to couple and then converge to the optimal value. In particular, estimates of node 1 and node 2 are kept closer each other since the first matrix of above is selected with higher probability. Similar results have been obtained with $\gamma(k) = \gamma \ll 1$.

The proofs of the convergence results in [3,4] share mainly the same outline. A key element is proving that the averaging component (the consensus) of the algorithm converges exponentially fast. More formally, under 6′, [3] proves that $\mathbf{A}_{(s)}^{(k)}$ surely converges to the matrix $\mathbf{J} = 1/M\mathbf{1}\mathbf{1}^T$, and that there are two positive constants C and β such that $\|\mathbf{A}_{(s)}^{(k)} - \mathbf{J}\|_\infty \leq C\beta^{k-s}$ for all $k \geq s$. Under 6″, [4] proves almost surely convergence of $\mathbf{A}_{(s)}^{(k)}$ to $\mathbf{J}$, and an exponential convergence rate in expectation, i.e. $E[\|\mathbf{A}_{(s)}^{(k)} - \mathbf{J}\|_\infty] \leq C\beta^{k-s}$. Then, similar bounds are established for the distance between $\mathbf{y}$ and $\mathbf{x}^*$, and between $\mathbf{x}^i$ and $\mathbf{y}$, and convergence results (when $\gamma(k)$ satisfies condition 5) and asymptotic bounds (when $\gamma(k)$ is constant) follow from the exponential convergence rate of the averaging component. As a consequence of the different kind of convergence for $\mathbf{A}_{(s)}^{(k)}$ in the two cases, these results hold surely under 6′ and almost surely under 6″.

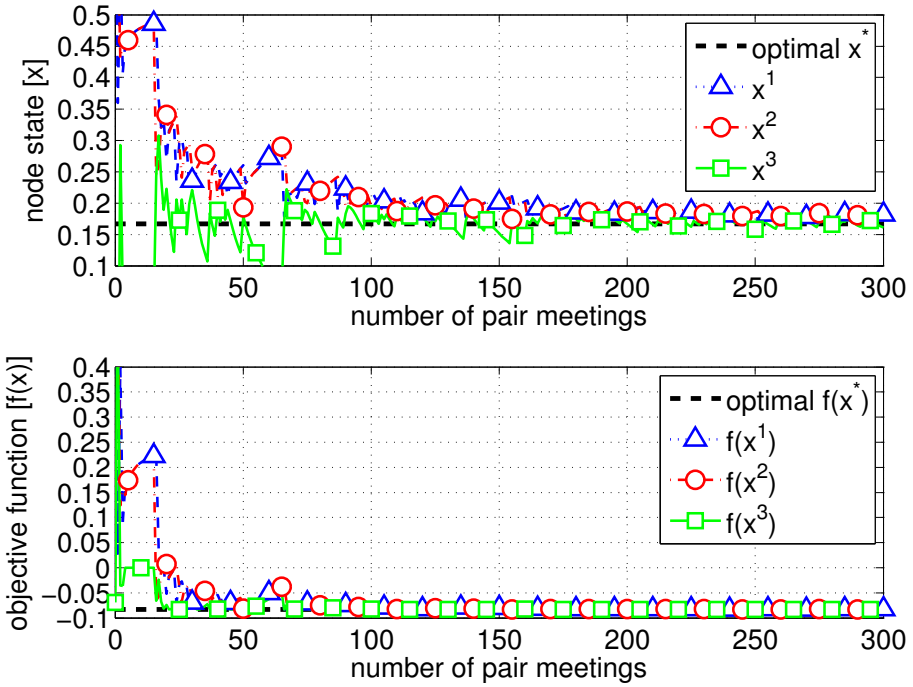

Figure 1: Toy example, convergence of the three estimates. Top graph: state's estimates for each node. Bottom graph: objective function value computed in the state's estimates of each node.

3 Application to Optimization in DTNs

The distributed sub-gradient method presented in Sec. 2 is particularly appealing in the context of Delay Tolerant Networks, see e.g., [8] and [9]. DTNs are sparse and/or highly mobile wireless ad hoc networks where no continuous connectivity guarantee can be assumed. This intrinsically leads to the impossibility of collecting, at low cost and at a single data processing point, the information needed to solve network optimization problems in a centralized fashion. Due to this, in the present report we advocate the use of distributed approaches, which lend themselves well to distributed and communication efficient optimization. To be more concrete, in what follows we briefly discuss two possible DTN scenarios where a global network's function f has to be optimized.

One central problem in DTNs is related to routing packets towards intended destinations. Common techniques, designed to overcome the absence of a complete route to the destination, rely on multi-copy dissemination of messages in the network [10]. In this context, it is natural to define global optimization functions that are able to take in account the trade-off between delivery time and the cost due to the use of resources such as buffer space, bandwidth and transmission power. Functions of this kind are convex and can be written as sum of locally measurable quantities, thus we can optimized them through the distributed sub-gradient framework [11].

A second DTN scenario concerns the dissemination of dynamic content, like news or traffic information. Referring to the application example of [7], we might think of a Service Provider (SP) with limited bandwidth, that has to decide the update rate to assign at each node. Nodes can share their content when they meet with the global objective of maintaining the average information in the network as "fresh" as possible. [7] shows that this problem can also be formalized as a classical convex optimization problem, and that the corresponding global objective function can be expressed in terms of the sum of local functions. The derivation of the latter local functions entails the collection of statistics which are computed at each node only considering its own meeting occurrences. As an application example for our techniques, in Sec. 6 we apply the distributed sub-gradient framework to this scenario.

From a general perspective, the distributed sub-gradient optimization in DTNs can be applied as follows. Nodes exchange their local estimates every time they meet and perform the update step in equation (1) at a given sequence of time instants $\{t_k\}_{k\geq 1}$. This sequence can either coincide with the meeting times, i.e., each time two nodes meets they exchange and subsequently update their estimates or be independent from them, i.e., in this case $\{t_k\}_{k\geq 1}$ is defined a priori and is known to every network node. In any case, the weight matrices $\mathbf{A}(k)$ originate from the node meeting process. In particular, we can consider the contact matrix $\mathbf{C}(k)$, where $c_{ij}(k) = 1$ if node i has met node j since last time instant t_{k-1}, and $c_{ij}(k) = 0$ otherwise. We denote as $\mathcal{C}$ the (finite) set of all possible $M \times M$ contact matrices describing the contacts among M nodes. Each node i can thus calculate its own weights $a_{ij}(k)$, for $j = 1, \ldots, M$, in one of the following two ways (which guarantee that the matrix $\mathbf{A}(k)$ is doubly stochastic):

Rule 1 (Updates independent from meetings) For $j \neq i$, set $a_{ij}(k) = 1/M$ if $c_{ij}(k) = 1$, otherwise set $a_{ij}(k) = 0$. Set $a_{ii}(k) = 1 - \sum_{j\neq i} a_{ij}(k)$. This method requires each node to know M, i.e., the total number of nodes in the system.

Rule 2 (Updates synchronized with meetings) Whenever node i meets node j, it also updates its estimate. In this case, set $a_{ij}(k) = a$ with $0 < a < 1$, $a_{ii}(k) = 1 - a$ and $a_{ih} = 0$ for $h \neq j, i$.

Next, we discuss two key issues that can negatively impact the convergence of the distributed optimization process in a DTN scenario. The first one is related to the validity of assumptions 6′ and 6″. In fact, condition 6′ is essentially equivalent to assume that there is a deterministic bound for the inter-meeting times of two nodes (that meet infinitely often), and this is for example not the case for all the random mobility models usually considered, see e.g., [12]. Condition 6″ relaxes 6′, but requires the independence of the meetings occurring in each time slot $[t_{k-1}, t_k]$, and meetings under realistic mobility are instead correlated (e.g., if in the recent past i has met j and j has met h, then the three nodes are likely to be close in space and the probability that i meets h is higher that with uniform and independent mobility). We address this issue in Sec. 4, where we prove that convergence results hold under more general assumptions on the stochastic process of the matrices $\mathbf{A}(k)$.

The second issue is related to the synchronicity of the updates. In fact, the original framework [3, 4] requires all the nodes to update their estimates at the

same time instants. This is not always feasible in a disconnected and distributed scenario like a DTN. For example, under Rule 2 the reader may have noted that synchronous updates require each node to know when a meeting between any two nodes in the network occurs. This does not appear to be practical. Under Rule 1, which requires each node to know the total number of nodes in the system, nodes should also try to keep their internal clocks synchronized in order to be able to perform their updates at "close enough" time instants and this presents some difficulties as well. We now show through an example that we cannot simply ignore the issue of synchronicity and that a direct application of the algorithm described in the previous section in general does not lead to correct results. Coming back to the toy example presented in Sec. 2, we observe that we can think our three matrices in (5) as generated according to Rule 2, when the meeting process has the following characteristics: at each time slot, node 1 and node 2 meet with probability 2/3, node 1 and 3 meet with probability 1/6 and node 2 and 3 meet with probability 1/6. Fig. 2 shows the evolution of the estimates when the step-size is constant and equal to $25 \cdot 10^{-4}$, both for the synchronous case, where all the nodes update their estimates when a meeting occurs (even the node that is not involved in the meeting), and for the asynchronous case, where only the nodes involved in the meeting perform the update. The curves represents the average estimates over 100 different simulations with different meeting sequences. We note that in the synchronous case (top figure) all nodes agree on the optimal value to set x, whereas, in the asynchronous case (bottom figure) the estimates still converge, but not to the correct point of minimum for the global function f. We address this issue in Sec. 5, where we understand the roots of this convergence problem and propose some simple modifications to the basic framework to effectively cope with it.

In our opinion, these extensions to the basic framework proposed in [3, 4], while motivated in this report by the DTN scenario, are of wide interest for other possible applications such as mobile wireless ad hoc and sensor networks.

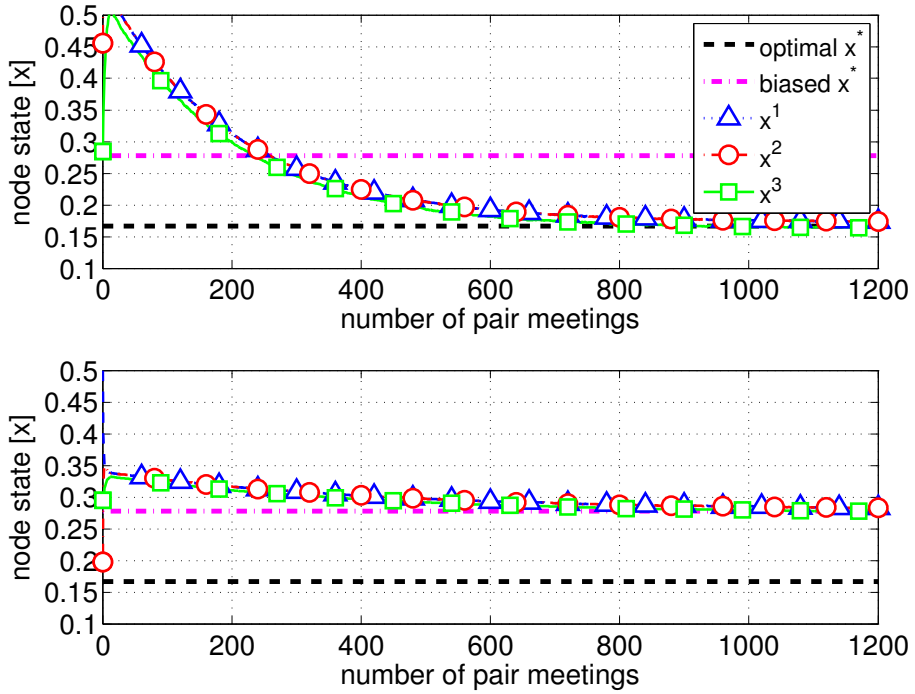

Figure 2: Toy example, convergence of the three estimates in case of fixed step-size $\gamma = 25 \cdot 10^{-4}$. Results have been averaged over 100 simulation runs. Top graph: synchronous updates. Bottom graph: asynchronous updates.

4 Extension to more General Mobility Models

In our DTN scenario, we consider that the weights are determined from the contacts through a bijective function (as in the case of the two rules presented in Sec. 3). Then conditions 5 and 6″ of [4], can be expressed in terms of the sequence of contact matrices as follows: the contact matrices $\mathbf{C}(k)$ are i.i.d. and $\mathrm{E}[\mathbf{C}(k)]$ is an irreducible aperiodic matrix. In this section, we extend the convergence results to the following, more general, mobility model.

Assumption 1 (Mobility model)**.** *It exists an irreducible, aperiodic and stationary Markov chain Φ with a finite or countable set of states S and a function $g : S \to \mathcal{C}$, such that $\mathbf{C}(k) = g(\Phi_k)$, for each $\Phi_k \in S$. Moreover, $E[\mathbf{C}(k)]$ is an irreducible aperiodic matrix.*

Since there is a bijective correspondence among weight and contact matrices, we observe that under assumption 1, it also exists a function $\hat{g} : S \to \mathcal{A}$, such that $\mathbf{A}(k) = \hat{g}(\Phi_k)$. The case when the contact matrices (and then the weight matrices) are i.i.d. is a particular case of our mobility model.

Our proof follows the same outline of [3, 4] presented in Sec. 2. The main issue is to prove the exponential rate of convergence of the backward product $\mathbf{A}_{(1)}^{(k)}$ to $\mathbf{J} = 1/M\mathbf{1}\mathbf{1}^T$.

Before proving the convergence of the backward product, we need to recall an ergodic property of the time shift operator θ for irreducible, aperiodic and stationary Markov chains. The definitions of measure-preserving and ergodic operators may be found in Appendix B (see also chapter V of [13] for more details).

Proposition 1. *Given an irreducible aperiodic and stationary Markov chain Φ with finite or countable states, the shift operator θ is measure-preserving and ergodic together with all its powers θ^k, where $k \in \mathbb{N}$.*

Proof. For stationary Markov chains the shift operator and its powers are measure-preserving by definition of stationarity. Moreover, irreducible, aperiodic and stationary Markov chains with finite or countable states are mixing (see Theorem 3.1 in [14]). From the definition of mixing, if θ is mixing, also the operator θ^k is mixing for any given $k \in \mathbb{N}$. But every mixing operator is also ergodic by Theorem 2 in [13], then θ^k is ergodic for any $k \in \mathbb{N}$. □

We observe that the stochastic process $\mathbf{A}(k) = \hat{g}(\Phi_k)$ is not in general a Markov chain, because different states of S may be mapped to the same weight matrix, but nevertheless it is stationary and ergodic.

We will also need the following result, whose proof is in Appendix C.

Lemma 1. *(Windowing a Markov chain) Let $\Phi = \{\Phi_n, n \in \mathbb{N}\}$ be an irreducible, aperiodic and stationary Markov chain. Consider the stochastic process $\Psi = \{\Psi_n, n \in \mathbb{N}\}$, where $\Psi_n = (\Phi_n, \Phi_{n+1}, \cdots, \Phi_{n+h-1})$ with h a positive integer. Ψ is also an irreducible aperiodic stationary Markov chain.*

Now we have all the instruments to study the convergence of $\mathbf{A}_{(1)}^{(k)}$. First, we prove the convergence to $\mathbf{J}$. This is a corollary of results in [15].

Proposition 2 (Convergence of the backward product)**.** *Let assumption 1 hold, then*

$$\lim_{k \to +\infty} \mathbf{A}_{(1)}^{(k)} = \frac{1}{M} \mathbf{1}\mathbf{1}^T \triangleq \mathbf{J} \quad \textit{almost surely (a. s.)} \ .$$

Proof. We observe that $\mathbf{A}(k)$ is a stationary and ergodic sequence of stochastic matrices with strictly positive diagonal entries. Moreover, $\mathrm{E}[\mathbf{A}(k)]$ is an irreducible aperiodic matrix, then its eigenvalue with the second largest module has module strictly smaller than 1 ($|\lambda_2(\mathrm{E}[\mathbf{A}(k)])| < 1$). From Theorem 3 in [15], it follows that, with probability one, for each sequence $\mathbf{A}(k)$ it exists a vector $\mathbf{v} \in \mathbb{R}^M$, such that $\sum_i v_i = 1$ and

$$\lim_{k \to +\infty} \mathbf{A}_{(1)}^{(k)} = \mathbf{1}\mathbf{v}^T \ .$$

Note that in general $\mathbf{v}$ is a random variable, depending on the specific sequence $\{\mathbf{A}(k)\}_{k \geq 1}$. The matrices $\mathbf{A}(k)$ are doubly stochastic, then $\mathbf{w} = 1/M\mathbf{1}$ is a left eigenvector corresponding to the unit eigenvalue for all the matrices $\mathbf{A}(k)$. Theorem 4 in [15], guarantees that in this case the above vector $\mathbf{v}$ is a deterministic constant almost surely and in particular is equal to $\mathbf{w}$. This concludes the proof. □

Now we are ready to prove that the convergence rate is almost always exponential.

Proposition 3. *Under assumption 1 on the mobility models, if the matrices are doubly stochastic, then for almost all the sequences there exist $C > 0$ and $0 < \beta < 1$ (with C in general depending of the sequence) such that for $k \geq s$*

$$\left\| \mathbf{A}_{(s)}^{(k)} - \mathbf{J} \right\|_{\max} \leq C\beta^{k-s} \ .$$

Proof. Given a matrix $\mathbf{A}$, consider the coefficient of ergodicity, see e.g., [16]:

$$\tau_1(\mathbf{A}) = \frac{1}{2} \max_{i,j} \sum_{s=1}^{M} \left| [\mathbf{A}]_{is} - [\mathbf{A}]_{js} \right| .$$

In the proof of Theorem 3 in [15] is shown that it exists a positive natural h and $\eta < 1$ such that

$$\mathbb{P}\left[\tau_1 \left(\mathbf{A}_{(s+(r-1)h)}^{(s+rh-1)} \right) < \eta \text{ for infinitely many r} \right] = 1 \ . \tag{6}$$

Then we decompose $\mathbf{A}_{(s)}^{(k)}$, in the product of i_k blocks of size h and one block of size $(k+1) \mod h$ as it follows:

$$\mathbf{A}_{(s)}^{(k)} = \mathbf{A}_{(s+i_k h)}^{(k)} \mathbf{A}_{(s+h(i_k-1))}^{(s+hi_k-1)} \cdots \mathbf{A}_{(s)}^{(s+h-1)} \ .$$

Because of the properties of a coefficient of ergodicity:

$$\tau_1 \left(\mathbf{A}_{(s)}^{(k)} \right) \leq \tau_1 \left(\mathbf{A}_{(s+i_k h)}^{(k)} \right) \prod_{j=1}^{i_k} \tau_1 \left(\mathbf{A}_{(s+h(j-1))}^{(s+hj-1)} \right) \leq$$

$$\leq \prod_{j=1}^{i_k} \tau_1 \left(\mathbf{A}_{(s+h(j-1))}^{(s+hj-1)} \right) \ .$$

Then, we can write:

$$\log \left(\tau_1 \left(\mathbf{A}_{(s)}^{(k)} \right) \right) \leq \sum_{j=1}^{i_k} \log \left(\tau_1 \left(\mathbf{A}_{(s+h(j-1))}^{(s+hj-1)} \right) \right) \ .$$

We now consider the Markov chain Φ, that "generates" the sequence of matrices $\mathbf{A}(k)$ underlying the mobility process. Because of Lemma 1, $\Psi_t = (\Phi_t, \Phi_{t+1}, \cdots, \Phi_{t+h-1})$ is an irreducible aperiodic stationary Markov chain, and then all the powers of the shift operator θ, and in particular θ^h, are ergodic. We observe that $\log \left(\tau_1 \left(\mathbf{A}_{(j)}^{(j+h-1)} \right) \right)$ is a function of $(\Phi_j, \Phi_{j+1}, \cdots, \Phi_{j+h-1})$ and then of Ψ_j. We call such function f, i.e., $f(\Psi_j) \stackrel{\text{def}}{=} \log \left(\tau_1 \left(\mathbf{A}_{(j)}^{(j+h-1)} \right) \right)$. Note that $f(\Psi_t) \leq 0$ and, from equation (6), $f(\Psi_t) < \log(\eta) < 0$ infinitely often almost surely. We can then apply Birkhoff's ergodic theorem to the random sequence $\{f(\Psi_s), f(\Psi_{s+h}), f(\Psi_{s+2h}), \cdots\}$, and it follows that:

$$\lim_{i \to \infty} \frac{1}{i} \sum_{j=1}^{i} \log \left(\tau_1 \left(\mathbf{A}_{(s+h(j-1))}^{(s+hj-1)} \right) \right) =$$

$$= \lim_{i \to \infty} \frac{1}{i} \sum_{j=1}^{i} f(\Psi_{s+hj}) = \mathrm{E}[f(\Psi_t)] < 0 \quad \text{a. s.} \ ,$$

therefore,

$$\limsup_{h\to\infty} \frac{1}{h} \log\left(\tau_1\left(\mathbf{A}_{(s)}^{(s+h)}\right)\right) \leq \mathrm{E}[f(\Psi_t)] < 0 \quad \text{a. s. .}$$

Consider $\mathrm{E}[f(\Psi_t)] < \zeta < 0$, then for almost all the sequences it exists h_0, such that for all $h \geq h_0$, it holds:

$$\frac{1}{h} \log\left(\tau_1\left(\mathbf{A}_{(s)}^{(s+h)}\right)\right) \leq \zeta, \text{ i.e., } \tau_1\left(\mathbf{A}_{(s)}^{(s+h)}\right) \leq e^{\zeta h} .$$

If we define $\beta = \exp(\zeta) < 1$, $C = \beta^{-h_0}$ and recall that $\tau_1\left(\mathbf{A}_{(s)}^{(k)}\right) \leq 1$, we obtain:

$$\tau_1\left(\mathbf{A}_{(s)}^{(k)}\right) \leq C\beta^{k-s} \text{ for } k \geq s . \tag{7}$$

In the above equation, the value of the constant h_0 depends on the specific random sequence and also on s (while the same value ζ can be selected for all the sequences and independently from s). We need then to use a corollary of the ergodic theorem about "nearly uniform" convergence that is stated as proposition (1.5) in [17]: if $f(\cdot)$ is square integrable, then for almost all the sequences we can select h_0 independently from s. Clearly this is the case for our function $f(\Psi_t)$, therefore we can conclude that C in (7) only depends on the considered sequence.

So far we have established the existence of a geometric convergence result for the ergodic coefficient τ_1. The last step of our proof requires us to prove a geometric bound for the distance between $\mathbf{A}_{(s)}^{(k)}$ and its almost sure limit $\mathbf{J}$. In particular we prove that

$$\left|\left[\mathbf{A}_{(s)}^{(k)}\right]_{uv} - \frac{1}{M}\right| \leq 2\tau_1\left(\mathbf{A}_{(s)}^{(k)}\right) , \tag{8}$$

for each u and v. From (8) a geometric bound follows also for $\|\mathbf{A}_{(s)}^{(k)} - \mathbf{J}\|_{\max}$. The derivation of (8) has no particular difficulty and mainly follows the proof of Theorem 4.17 in [16], then we have moved it to Appendix D. □

In [4] a different result is proven, i.e., that there exist $\hat{C}$ and $\hat{\beta}$ such that

$$\mathrm{E}\left[\left\|\mathbf{A}_{(s)}^{(k)} - \mathbf{J}\right\|_{\max}\right] \leq \hat{C}\hat{\beta}^{k-s} .$$

Then a series of inequalities for the expected values of $\|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2$ are obtained for all i. Using Fatou's Lemma, along with the non-negativeness of distances, it is possible to derive inequalities that hold with probability 1. Using Proposition 3, instead, it is possible to obtain the same inequalities directly without the need to consider the expectation.

5 Asynchronous Updates

In this section, we study how the presented framework needs to be extended in order to support the case when nodes asynchronously update their status. We consider the sequence $\{t_k\}_{k\geq 1}$ of time instants at which one or more nodes perform an update of their estimates. Again, we denote that the estimate of node i at time t_k (immediately before the update) is $\mathbf{x}^i(k)$ and represent all the estimates through the matrix $\mathbf{X}(k)$. The evolution of the estimates can still be expressed in a matrix form similarly to (2):

$$\mathbf{X}(k+1) = \mathbf{A}(k)\mathbf{X}(k) - \mathbf{\Gamma}(k)\mathbf{D}(k) \ , \tag{9}$$

where $\mathbf{\Gamma}(k)$ is a diagonal matrix and the element $[\mathbf{\Gamma}(k)]_{ii}$ is simply the step-size used by node i at the k-th update. We denote this step-size as $\gamma_i(k)$. If j is not among the nodes which perform the update at time t_k, then it will simply be $a_{jj}(k) = 1$, $a_{jh}(k) = 0$ for $h \neq j$ and $\gamma_{jj}(k) = 0$.

In what follows we first consider the case of decreasing step-sizes, similarly to condition 1, described in Sec. 2. That is, we will consider that for each i, the sequence $\{\gamma_i(k)\}_{k\geq 1}$ satisfies: $\sum_{k=1}^{\infty}\gamma_i(k) = \infty$ and $\sum_{k=1}^{\infty}\gamma_i(k)^2 < \infty$.

We can go over the rationale in [4] and prove similar results for the new system description. In particular, our proof of the exponential convergence rate of $\mathbf{A}_{(s)}^{(k)}$ holds clearly also in this case. Bounds for the distance between $\mathbf{x}^i(k)$ and $\mathbf{y}(k) = 1/M\mathbf{1}^T\mathbf{X}(k)$ hold with minimal changes, so that we can prove the analogous of Proposition 2 in [4]:

Proposition 4 (Convergence of Agent Estimates)**.** *Under assumption 1, the estimate of each node converges almost surely to the vector* $\mathbf{y}(k)$*, i.e.,*

$$\lim_{k\to+\infty} \|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 = 0 \quad a.\ s.\ , \textit{for all } i\ .$$

Proof. See Appendix E. □

The following step is to use bounds for the distance between $\mathbf{y}(k)$ and $\mathbf{x}^*$ (a point of minimum for f) to show that $\lim_{k\to+\infty}\mathbf{y}(k) = \mathbf{x}^*$.

In particular the following inequality is derived in [4] (for the synchronous case they are considering):

$$\sum_{s=1}^{k}\gamma(s)\left[f(\mathbf{y}(s)) - f(\mathbf{x}^*)\right] \leq \frac{M}{2}\|\mathbf{y}(1) - \mathbf{x}^*\|_2^2 + \\ + 2L\sum_{j=1}^{M}\sum_{s=1}^{k}\gamma(s)\|\mathbf{y}(s) - \mathbf{x}^j(s)\|_2 + \frac{L^2}{2}\sum_{s=1}^{k}\gamma^2(s)\ .$$

The first term at the right hand side of the above inequality is a constant, the last term is summable because of the assumption on the step-sizes. In [4] it is proven that $\sum_{s=1}^{\infty}\gamma(s)\|\mathbf{y}(s) - \mathbf{x}^j(s)\|_2 < \infty$ almost surely. Thus they show that

$$0 \leq \sum_{s=1}^{\infty}\gamma(s)\left[f(\mathbf{y}(s)) - f(\mathbf{x}^*)\right] < \infty \quad \text{a. s.}\ , \tag{10}$$

from (10) and the fact that $\sum_{s=1}^{\infty} \gamma(s) = \infty$, it is possible to conclude that

$$\liminf_{k\to\infty} f(\mathbf{y}(k)) = f(\mathbf{x}^*) \quad \text{a. s. , and } \lim_{k\to\infty} \mathbf{x}^i(k) = \mathbf{x}^* \text{ a. s. .}$$

In Appendix F, a similar derivation is carried on, leading to the following generalization of (10):

$$\sum_{s=1}^{\infty} \sum_{i=1}^{M} \gamma_i(s) \left[f_i(\mathbf{y}(s)) - f_i(\mathbf{x}^*) \right] < \infty \quad \text{a. s. .} \tag{11}$$

Unfortunately, the different values of $\gamma_i(s)$ do not allow us to formulate the inequality above in terms of the global function f as in (10).

We do not have currently a formal result stating under which conditions the asynchronous system converges to the optimal solution, but (11) suggests us that all the weights $\gamma_i(k)$ should have on average the same value. We then propose the following conjecture, that we support later with some examples:

Conjecture 1. *When updates are asynchronous, convergence results for sub-gradient methods hold if* $E[\gamma_i(k)] = E[\gamma_j(k)]$ *for each* i *and* j.

Let us see how we can guarantee this condition in different cases. We consider that updates occur after every meeting following rule 2 in Sec. 3. Moreover consider that $\gamma_i(k) = 1/n_i(k)$, where $n_i(k)$ is the total number of updates node i has performed until the time instant t_k. If the meeting process follows a Poisson process with total rate λ and at each instant the probability that node i meets another node is p_i, we expect that by time k, node i has $p_i k$ meetings (and an equal number of updates). Then the expected value of its step-size is $\mathrm{E}[\gamma_i(k)] = \mathrm{E}[1/n_i(k)] = p_i/(p_i k) = 1/k$. In conclusion if step-sizes follow the rule $\gamma_i(k) = 1/n_i(k)$, we expect the asynchronous sub-gradient mechanism to converge to the optimal solution. Fig. 3 (top figure) shows that this is true for our toy example. The simulations for the optimization problem considered in Sec. 6 confirm such convergence.

Let us now revisit the example in Sec. 3 showing that the estimates were not converging to a point of minimum (Fig. 2, bottom figure). Here step-sizes were constant, i.e. $\gamma_i(k) = \gamma$. Now, reasoning as above we can conclude that $\mathrm{E}[\gamma_i(k)] = p_i\gamma$. Hence the expected values are not equal as far as node meeting rates (and then update rates) are not equal: this was the case of our example, where[3] $p_1 = 5/6$, $p_2 = 5/6$ and $p_3 = 1/3$. Intuitively, we expect convergence to be biased towards values closer to the optimum of the local functions of those nodes that perform the updates more often. Equation (11) suggests us that what the distributed mechanism was really doing is to minimize the function $\sum_i p_i f_i = (3/2)x^2 - (5/6)x$ rather then $f = \sum_i f_i = 3x^2 - x$. This is the case, being that the estimates are converging to 5/18 (dot-dashed line in all the previous figures). If now we want to correct the bias, it is sufficient to consider that each node selects its step-size inversely proportional to its meeting rate. Fig. 3 (bottom figure) shows that also this correction leads the estimates to converge to the correct results.

[3] Note that meetings always involve two nodes, this is the reason why $p_1 + p_2 + p_3 = 2$.

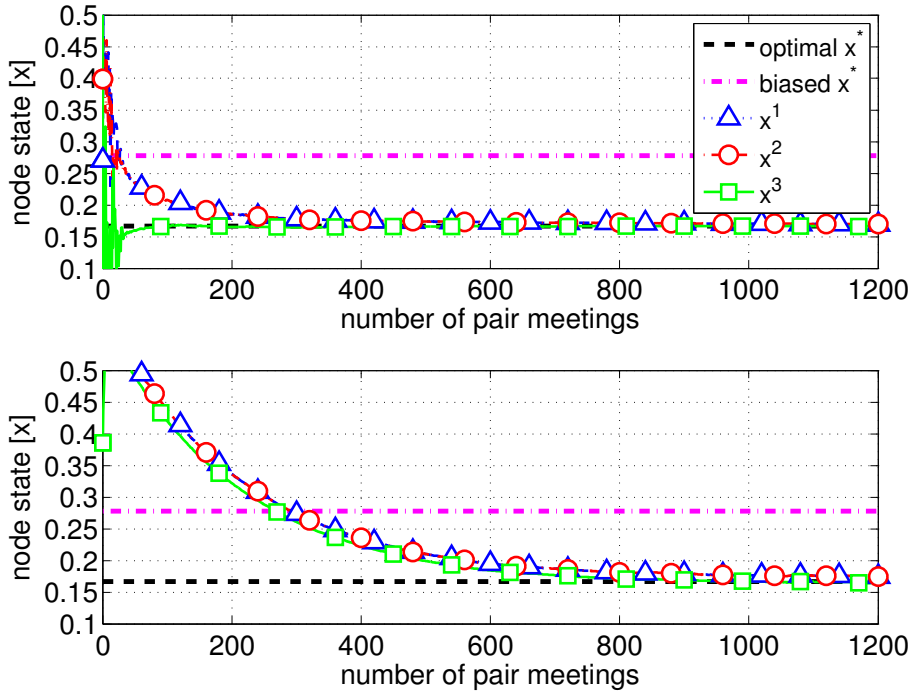

Figure 3: Toy example, convergence of the three estimates in case of asynchronous updates. Results have been averaged over 100 simulation runs. Top graph: decreasing step-size $\gamma_i(k) = 1/n_i(k)$. Bottom graph: weighted fixed step-size $\gamma_i(k) = p_i^{-1} \cdot 25 \cdot 10^{-4}$.

6 Application in DTNs: a Case Study

In this section we apply the distributed sub-gradient method with our enhancements to a DTN scenario inspired by the work in [7]. As explained in Sec. 5 our enhancements consist of: 1) allowing nodes to update asynchronously their estimates, i.e., whenever any two of them meet and 2) applying the decreasing step size rule to avoid possible bias effects in the convergence towards the global optimum.

All the nodes in the network are interested in the same dynamic information content and can share it whenever they meet. The information update is performed by a Service Provider (SP) that injects fresh information in the network according to a Poisson process of parameter μ update/sec. At a given instant $\bar{t}$ we call $t_i(\bar{t})$ the time at which the SP generated the most recent content version available at node i, then $Y_i(\bar{t}) = \bar{t} - t_i(\bar{t})$ is the age of such version. An information content has a non-increasing value in time. For example, we give value $u_i(Y_i(\bar{t}))$ to the information stored in node i, where $u_i(\cdot)$ is a non-increasing function. The goal of the SP is to optimize

$$f(\mathbf{x}) = \sum_{i=1}^{M} f_i(\mathbf{x}) = \sum_{i=1}^{M} E_{\mathbf{x}}[u_i(Y_i)] \ , \tag{12}$$

where $\mathbf{x} \in \mathbb{R}^M$ is the rate allocation vector, such that $\sum_{i=1}^{M} x_i \leq \mu$ and $x_i \geq 0$ for all i. Note that the age $Y_i(\bar{t})$ is modeled as a random variable Y_i depending on $\mathbf{x}$. In [7], equation (12) is proved to be concave and therefore the optimal $\mathbf{x}$ can be obtained by the SP using standard optimization techniques (see e.g., [18]) such as the projected gradient descent algorithm[4]: namely, iteratively computing $\mathbf{x}(k+1) = \Pi\left(\mathbf{x}(k) + \gamma_k \nabla f(\mathbf{x}(k))\right)$, where $\{\gamma_k\}_{k\geq 0}$ is a positive sequence of parameters such that $\sum_k \gamma_k = \infty$, $\lim_{k\to\infty} \gamma_k = 0$ and Π is the projection onto the feasible set for $\mathbf{x}$. In general, a closed formula for $f(\mathbf{x})$ is not known; thus, the gradient needs to be estimated as explained in [7]. Here, our purpose is to focus on the distributed sub-gradient method so we consider the specific case where updates can travel at most two hops, thus avoiding to address the gradient estimation's issue. In detail, at a given instant $\bar{t}$, let us call $t_j^{SP}(\bar{t})$ the time at which the SP directly injected fresh content to node j, then we define the following protocol's rule:

Definition 1 (Content Sharing)**.** When a node j meets a node i at time $\bar{t}$, j will copy to user i the last content downloaded directly from the SP if this content is more recent then the content stored in i, i.e., $t_i(\bar{t}) < t_j^{SP}(\bar{t})$.

In this case, assuming also that a) $u(Y_i) = \chi\{Y_i \leq \tau\}$ for all nodes i, where τ is a given threshold after which the information is worthless[5] (e.g., the information consists of news about events that expire after some time) and b) the meeting process among node pairs is Poisson distributed, we can compute the local utility function for each node as

$$f_i(\mathbf{x}) = 1 - \left[\prod_{j\in\mathcal{N}_i} \frac{x_j e^{-\lambda_{ij}\tau} - \lambda_{ij} e^{-x_j\tau}}{x_j - \lambda_{ij}}\right] e^{-x_i\tau} \ , \tag{13}$$

where λ_{ij} is the meeting rate between i and j and $\mathcal{N}_i \stackrel{\text{def}}{=} \{j : \lambda_{ij} > 0\}$. The global utility function in (12) is then simply obtained summing (13) over $i = 1, 2, \ldots, M$. The computations to derive (13) can be found in Appendix G.

The local gradient function needed in (1) can be computed directly from (13), where nodes only need to estimate simple statistics on their own meeting rates. Clearly, (12) can be optimized also in a centralized fashion collecting, for example at the SP itself, information about the statistic of the overall network meeting process [7]. However, in a DTN scenario the SP may be able to communicate with a group of connected nodes only for short periods of time, that we would like to exploit transmitting the actual content users care about. Moreover, issues related to privacy easily apply to this scenario: for example, a node may prefer not to disclose information about its meetings to the SP and, in some cases, it would be equally desirable to maintain information about the utility function $u_i(\cdot)$ reserved (e.g., in military applications). A distributed approach is therefore of actual interest not only for DTNs, but also for scenarios that go beyond them.

To optimize $f(\mathbf{x})$ in a distributed fashion we can use the framework presented in Sec. 2. Each node i can compute a local estimate of the optimal allocation $\mathbf{x}$, i.e., $\mathbf{x}^i$, through iterative updates. In detail, when two nodes i and j meet

[4] For a discussion about the projected gradient method implemented in a distribute fashion see [19].

[5] $\chi\{y \leq \tau\}$ is equal to 1 if $y \leq \tau$ and 0 otherwise.

they: i) update $\mathbf{x}^i$ and $\mathbf{x}^j$ as in (1) and, ii) project the result so obtained onto the feasible set $\sum_{l=1}^{M} x_l \leq \mu$ and $x_l \geq 0$ for all $l \in \{1, \ldots, M\}$.

In our implementation of sub-gradient optimization all the $\mathbf{x}^i$ eventually converge to the optimum $\mathbf{x}^*$ of (12). Henceforth, the SP can retrieve the optimal transmission rates using the following "push-policy". During the execution of the algorithm each node i maintains its own estimate $\mathbf{x}^i$. The SP collects $\mathbf{x}^i$ from every node and obtains the rate allocation vector as $\mathbf{x} = (\sum_{i=1}^{M} \mathbf{x}^i)/M$.

To test the performance achievable by the distributed sub-gradient method under traces with memory, we simulated meeting events among $M = 10$ nodes as follows. Calling R_1 and R_2 two distinct regions of the space, nodes can be placed either in R_1 or in R_2. Only nodes that are within the same region can communicate with each other. We let nodes free to change region of placement according to a Poisson process of overall rate $\lambda_d = 0.1$, thus network's full connectivity is guaranteed. In addition, according to a Poisson process of parameter $\lambda_m = 1$ (note that $\lambda_m > \lambda_d$), we generate meeting events among pair of nodes belonging to the same region. Each node is selected for a meeting according to a weight which is proportionally inverse to its index, i.e., node i is selected with weight $w_i = i^{-3}$. Note that we generate a meeting process that is both stationary and ergodic, and along this process nodes have diverse contact rates. In particular node 1 has the highest contact rate, whilst node 10 the lowest. To sum up, letting nodes i and j update their states at their meeting times according to the above process, we obtain a corresponding sequence $\{\mathbf{A}(k)\}_{k\geq 1}$ that can be viewed as generated from an ergodic and stationary Markov chain. Also, given that nodes have different contact rates and asynchronous updates are performed, we know that a direct application of the distributed sub-gradient algorithm may lead to sub-optimal results, see the toy example of Sec. 3.

Figures 4–5 show simulation results for the above setting of parameters and $\tau = 20$.

Fig. 4 shows the optimal allocation rate for each user. When the bandwidth that the SP can use to send updates is very low (i.e., small μ), the best solution is that the SP uniquely sends updates to the node that has the higher contact rate, i.e., node 1; for large values of μ, instead, the SP can evenly send updates to all the nodes in the network. Interestingly, as already observed in [7], for some values of μ (in our case μ around $10^{0.7}$ update/sec) the optimal choice is for the SP to allocate more bandwidth (i.e., a larger fraction of μ) to the node with the lowest contact rate, namely, node 10. For these values of μ, in fact, those nodes with a large contact rate such as node 1 are able to maintain high values for their utility functions just by collecting information from the large number of nodes they meet.

In Fig. 5 we show the mean trajectory towards the optimal for two elements in $\mathbf{x} = (\sum_{i=1}^{M} \mathbf{x}^i)/M$, where the vectors $\mathbf{x}^i$ have been obtained along a sequence of $5 \cdot 10^4$ meetings considering $\mu = 10^{-1.1}$ update/sec. We note that the estimates provided through the distributed sub-gradient method converge to the theoretical optimal allocation values in Fig. 4. Concerns about the convergence rate of such estimates are out of the scope of the present report and will be addressed in the future research[6].

[6] Note, however, that a wide literature addressing this issue already exists. E.g., for the problem of designing suitable sequences $\{\mathbf{A}(k)\}$ to speed up the convergence of consensus see [20].

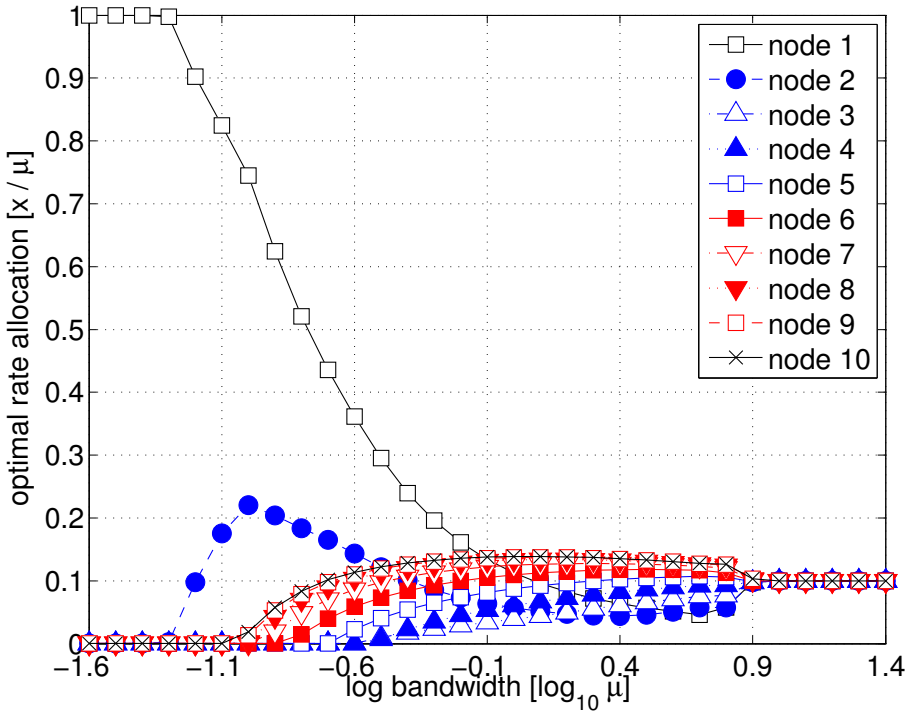

Figure 4: Optimal bandwidth allocation for a network of 10 nodes.

Finally, in Fig. 6 we draw with a solid-line the maximum of $f(\mathbf{x})$ corresponding to the optimal rate allocations in Fig. 4, which was obtained using a centralized solver [18]. Note that with illimited bandwidth, the maximum of $f(\mathbf{x})$ is equal to 10, i.e., when each node i has utility $u_i = 1$. For eight different values of the available bandwidth μ, we also plot with squared points the utility function values corresponding to the optimal rate allocation achieved again with $\mathbf{x} = (\sum_{i=1}^{M} \mathbf{x}^i)/M$ using sub-gradient optimization together with our enhancements. With crosses we show the performance of the sub-gradient optimization with a fixed step size [3], which neglects the asynchronous update issue. As expected, the results of the latter algorithm are sub-optimal. Most importantly, the solutions achieved with our approach are very close to the actual optimum for all values of μ. This confirms the validity of the distributed framework that we presented.

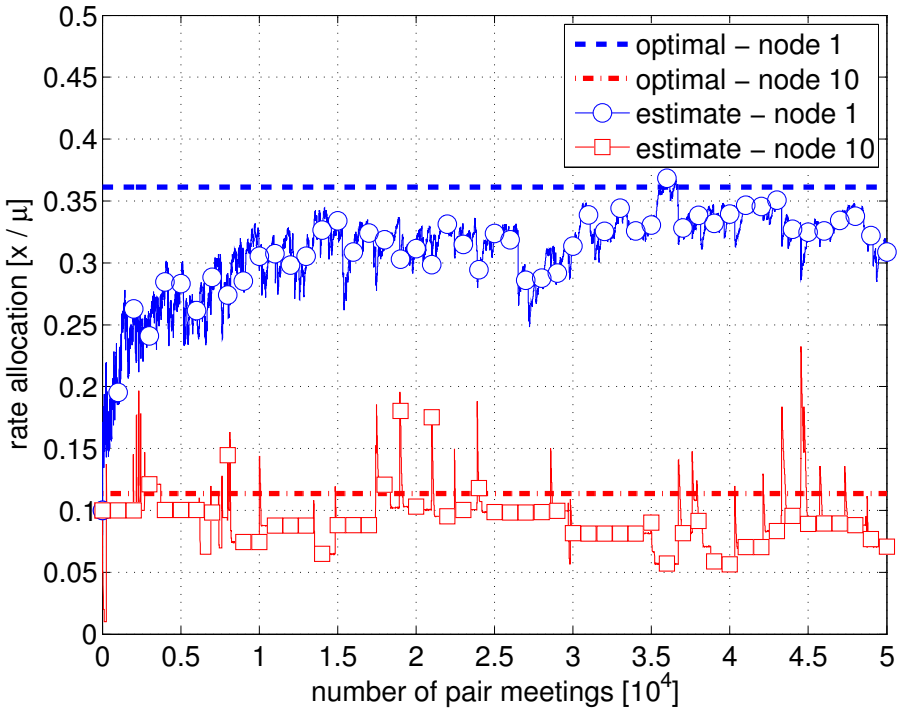

Figure 5: Example of convergence of estimate for two nodes when the sub-gradient method is used. Asynchronous updates and decreasing step-size.

7 Conclusions

In this report we considered the recent framework of the distributed sub-gradient optimization proposed in [3], and later extended in [4] for application on random scenarios. We pointed out that existing convergence results for this framework can be applied to DTNs only in the case of synchronous node operation and in the presence of simple mobility models without memory. Therefore, we addressed both these issues: first, we proved convergence to optimality of the sub-gradient optimization technique under a more general class of mobility processes and second, we proposed some modifications to the original sub-gradient algorithm so as to avoid bias problems (i.e., consisting of the convergence towards sub-optimal solutions) when nodes operate asynchronously. Finally, as a case study, we applied the presented framework to the optimization of the dissemination of dynamic content in a DTN. All the provided results confirmed that the distributed sub-gradient method is an effective and very promising tool for optimization in distributed contexts.

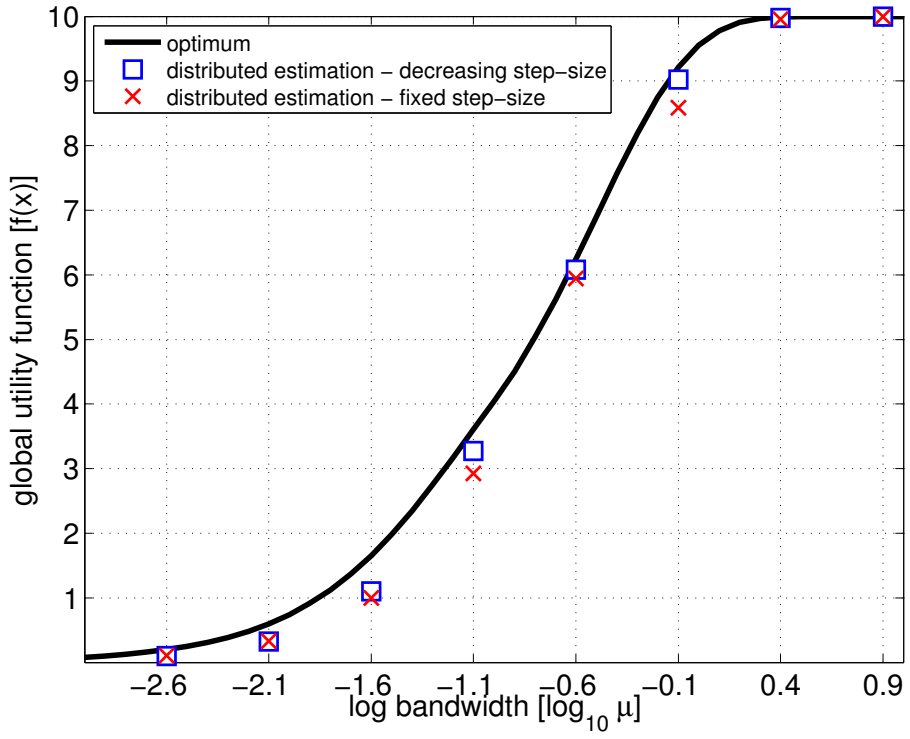

Figure 6: Performance comparison: centralized solver *vs* distributed method.

8 Acknowledgments

Authors kindly thank Dr. Michele Rossi, for his precious comments and suggestions.

A Derivation of Equation (2)

In this Appendix we show the mathematical steps that lead from equation (2) to equation (3).

From (2) we have that

$$\begin{aligned}\mathbf{X}(k+2) &= \mathbf{A}(k+1)\mathbf{X}(k+1) - \mathbf{\Gamma}(k+1)\mathbf{D}(k+1) = \\ &= \mathbf{A}(k+1)[\mathbf{A}(k)\mathbf{X}(k) - \mathbf{\Gamma}(k)\mathbf{D}(k)] - \mathbf{\Gamma}(k+1)\mathbf{D}(k+1) = \\ &= \mathbf{A}_{(k)}^{(k+1)}\mathbf{X}(k) - \mathbf{A}_{(k+1)}^{(k+1)}\mathbf{\Gamma}(k)\mathbf{D}(k) - \mathbf{\Gamma}(k+1)\mathbf{D}(k+1)\ ,\end{aligned}$$

where $\mathbf{\Gamma}(k)$ is a diagonal matrix and the element $[\mathbf{\Gamma}(k)]_{ii} = \gamma(k)$, for all i. Iteratively, replacing $k+2$ with $k+s+1$, $k+1$ with $k+s$ and k with $k+s-1$, respectively, leads to

$$\begin{aligned}&\mathbf{X}(k+s+1) = \\ &= \mathbf{A}_{(k+s-1)}^{(k+s)}\mathbf{X}(k+s-1) - \mathbf{A}_{(k+s)}^{(k+s)}\mathbf{\Gamma}(k+s-1)\mathbf{D}(k+s-1) + \\ &\quad - \mathbf{\Gamma}(k+s)\mathbf{D}(k+s) = \\ &= \mathbf{A}_{(k+s-1)}^{(k+s)}\left[\mathbf{A}(k+s-2)\mathbf{X}(k+s-2) - \mathbf{\Gamma}(k+s-2)\mathbf{D}(k+s-2)\right] + \\ &\quad - \mathbf{A}_{(k+s)}^{(k+s)}\mathbf{\Gamma}(k+s-1)\mathbf{D}(k+s-1) - \mathbf{\Gamma}(k+s)\mathbf{D}(k+s) = \\ &= \mathbf{A}_{(k+s-2)}^{(k+s)}\mathbf{X}(k+s-2) + \\ &\quad - \mathbf{A}_{(k+s-1)}^{(k+s)}\mathbf{\Gamma}(k+s-2)\mathbf{D}(k+s-2) + \\ &\quad - \mathbf{A}_{(k+s)}^{(k+s)}\mathbf{\Gamma}(k+s-1)\mathbf{D}(k+s-1) - \mathbf{\Gamma}(k+s)\mathbf{D}(k+s) = \\ &= \mathbf{A}_{(k+s-2)}^{(k+s)}\mathbf{X}(k+s-2) - \sum_{l=0}^{1}\mathbf{A}_{(k+s-l)}^{(k+s)}\mathbf{\Gamma}(k+s-1-l)\mathbf{D}(k+s-1-l) + \\ &\quad - \mathbf{\Gamma}(k+s)\mathbf{D}(k+s) = \\ &\qquad\vdots \\ &= \mathbf{A}_{(1)}^{(k+s)}\mathbf{X}(1) - \sum_{l=0}^{k+s-2}\mathbf{A}_{(k+s-l)}^{(k+s)}\mathbf{\Gamma}(k+s-1-l)\mathbf{D}(k+s-1-l) + \\ &\quad - \mathbf{\Gamma}(k+s)\mathbf{D}(k+s)\ ,\end{aligned}$$

and finally, replacing $k+s$ with k, we have that

$$\mathbf{X}(k+1) = \mathbf{A}_{(1)}^{(k)}\mathbf{X}(1) - \sum_{l=0}^{k-2}\mathbf{A}_{(k-l)}^{(k)}\mathbf{\Gamma}(k-1-l)\mathbf{D}(k-1-l) - \mathbf{\Gamma}(k)\mathbf{D}(k) \quad (14)$$

or, replacing in equation (14) $k-l$ with s

$$\mathbf{X}(k+1) = \mathbf{A}_{(1)}^{(k)}\mathbf{X}(1) - \sum_{s=2}^{k}\mathbf{A}_{(s)}^{(k)}\mathbf{\Gamma}(s-1)\mathbf{D}(s-1) - \mathbf{\Gamma}(k)\mathbf{D}(k)\ ,$$

that is exactly (3), as we wanted.

B Stationarity and Ergodicity: Concepts

Let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability spaces. Let also the function $T : \Omega \to \Omega$ be a measurable transformation of Ω into itself.

Definition 2 (Measure-Preserving)**.** A measurable transformation $T : \Omega \to \Omega$ into Ω is measure-preserving if, for every B in $\mathcal{F}$, $\mathbb{P}(TB) = \mathbb{P}(B)$.

Definition 3 (Stationarity)**.** A random sequence $\omega \in \Omega$ is stationary if the shift operator is measure preserving.

Definition 4 (Invariant set)**.** If T is a measure-preserving transformation, $B \in \mathcal{F}$ is an invariant set if $TB = B$, or equivalently $\mathbb{P}\left[(B \setminus TB) \cup (TB \setminus B)\right] = 0$.

Definition 5 (Ergodicity)**.** A measure-preserving transformation T is ergodic, if, given any invariant set $B \in \mathcal{F}$, it holds $\mathbb{P}(B) = 0$ or $\mathbb{P}(B) = 1$.

Definition 6 (Mixing)**.** A measure-preserving transformation T is mixing if, for all B and C in $\mathcal{F}$,

$$\lim_{n \to \infty} \mathbb{P}(B \cap T^n C) = \mathbb{P}(B)\mathbb{P}(C).$$

Note that when a random sequence is said to be ergodic tout court, it means that the shift operator is ergodic. Similarly when a random sequence is said to be mixing (or mixing in the ergodic-theoretic sense), it means that the shift operator is mixing.

C Proof of Lemma 1

In this Appendix we prove lemma 1, reported in the following for reader convenience.

Lemma. *(Windowing a Markov chain) Let $\Phi = \{\Phi_n, n \in \mathbb{N}\}$ be an irreducible, aperiodic and stationary Markov chain. Consider the stochastic process $\Psi = \{\Psi_n, n \in \mathbb{N}\}$, where $\Psi_n = (\Phi_n, \Phi_{n+1}, \cdots, \Phi_{n+h-1})$ with h a positive integer. Ψ is also an irreducible aperiodic stationary Markov chain.*

Proof. First of all it is evident that Ψ is also a Markov chain, whose states are possible h-uples of states of Φ, e.g $(s_1, s_2, \cdots, s_h)$. The transition probabilities could be calculated starting from those of Φ. Stationarity of Ψ easily follows from the stationarity of Ψ. Ψ is also irreducible because Φ is irreducible. In fact, given two states $s' = (s_1, s_2, \cdots, s_h)$ and $t' = (t_1, t_2, \cdots, t_h)$, for the irreducibility of Φ, it exists n_0, such that the chain Ψ moves from s' to a state $u' = (u_1, u_2, u_{h-1}, t_1)$ after n_0 steps, and then it is possible to move from s_h to t_1. t' is a state of Ψ and therefore it is also a valid sequence of state transitions for Φ, consequently in $h-1$ time steps, Φ can move from t_1 to t_h going through t_2, $\cdots$, t_{h-1} and Ψ can move from u' to t'. In conclusion in $n_0 + h - 1$ steps, Ψ can move from s' to t'.

Aperiodicity requires a more detailed discussion. Given a possible state $s' = (s_1, s_2, \cdots, s_h)$, we want to prove that the greatest common divisor of the possible time steps after which the chain Ψ can return in s' is equal to 1. Note that even if Φ had the property that it is possible to directly move from each state to itself, for a state s' with $s_i \neq s_1$ for some $i = 2, \cdots, h$, at least h steps are required to return to that state. Consider the minimum number k_0 of time steps after which the chain Φ can move from s_h to s_1 (again k_0 exists because Φ is irreducible). Consider then the increasing sequence of all the possible time steps $k_1, k_2, \ldots$ after which it is possible to return in s_1. Observe that also $2k_i$ belongs to this sequence. It is clear that Ψ can return in s' after $k_0 + k_1 + h - 1, k_0 + k_2 + h - 1, \ldots$ steps. Let us denote g the greatest common divisor of this sequence of numbers, we have that for each $i > 0$ $(k_0 + k_i + h - 1) \mod g = 0$. In particular also $(k_0 + 2k_i + h - 1) \mod g = 0$, and it follows that $(k_1) \mod g = 0$. This implies that g is also a divisor of the sequence $k_1, k_2, \ldots$. Since Φ is aperiodic, it follows that $g = 1$. This concludes the proof that Ψ is also aperiodic. □

D Proof of Equation (8)

In this Appendix we prove equation (8), namely

$$\left| \left[\mathbf{A}_{(s)}^{(k)} \right]_{uv} - \frac{1}{M} \right| \leq 2\tau_1 \left(\mathbf{A}_{(s)}^{(k)} \right) \ .$$

Proof. First we observe that a geometric bound holds also for the difference between any two elements on the same column and different rows. In fact, from the definition of $\tau_1(\cdot)$:

$$\left| \left[\mathbf{A}_{(s)}^{(k)} \right]_{u,v} - \left[\mathbf{A}_{(s)}^{(k)} \right]_{w,v} \right| \leq 2\tau_1 \left(\mathbf{A}_{(s)}^{(k)} \right) \ .$$

We call ϵ the right side of the above expression that we can rewrite as:

$$\left[\mathbf{A}_{(s)}^{(k)} \right]_{u,v} - \epsilon \leq \left[\mathbf{A}_{(s)}^{(k)} \right]_{w,v} \leq \left[\mathbf{A}_{(s)}^{(k)} \right]_{u,v} + \epsilon \ ,$$

therefore, for the double stochasticity of $\mathbf{A}(k)$, for all k

$$\begin{aligned} &\sum_{w=1}^{M} \left[\mathbf{A}(k+1) \right]_{z,w} \left(\left[\mathbf{A}_{(s)}^{(k)} \right]_{u,v} - \epsilon \right) \leq \\ &\leq \sum_{w=1}^{M} \left[\mathbf{A}(k+1) \right]_{z,w} \left[\mathbf{A}_{(s)}^{(k)} \right]_{w,v} \leq \\ &\leq \sum_{w=1}^{M} \left[\mathbf{A}(k+1) \right]_{z,w} \left(\left[\mathbf{A}_{(s)}^{(k)} \right]_{u,v} + \epsilon \right) \ , \end{aligned}$$

which is equal to

$$\left[\mathbf{A}_{(s)}^{(k)} \right]_{u,v} - \epsilon \leq \left[\mathbf{A}_{(s)}^{(k+1)} \right]_{z,v} \leq \left[\mathbf{A}_{(s)}^{(k+1)} \right]_{u,v} + \epsilon \ .$$

By induction:

$$\left[\mathbf{A}_{(s)}^{(k)} \right]_{u,v} - \epsilon \leq \left[\mathbf{A}_{(s)}^{(k+r)} \right]_{z,v} \leq \left[\mathbf{A}_{(s)}^{(k+1)} \right]_{u,v} + \epsilon \ ,$$

and letting r go to infinity

$$\left[\mathbf{A}_{(s)}^{(k)} \right]_{u,v} - \epsilon \leq \frac{1}{M} \leq \left[\mathbf{A}_{(s)}^{(k+1)} \right]_{u,v} + \epsilon \ ,$$

i.e.,

$$\left| \left[\mathbf{A}_{(s)}^{(k)} \right]_{u,v} - \frac{1}{M} \right| \leq \epsilon \ ,$$

and being that this inequality is true for all u and all v, a geometric bound can be derived also for $\left\| \mathbf{A}_{(s)}^{(k)} - \mathbf{J} \right\|_{\max}$. □

E Proof of Proposition 4

In this Appendix we prove Proposition 4, reported in the following for reader convenience. Since we proved Proposition 3, the result here presented (along with Proposition 5 in Appendix F) can be viewed as an easy extension of Theorem 1 in [4] to the case of mobility process obtained according to assumption 1.

Let us first to explicitly formalize the following assumption, according to both [3] and [4]:

Assumption 2 (Bounded Sub-gradients)**.** *Given $\mathbf{x} \in \mathbb{R}^N$, consider the sub-gradients of all the nodes in the network computed in $\mathbf{x}$, i.e., $\mathbf{d}^1, \ldots, \mathbf{d}^i, \ldots, \mathbf{d}^M$. There exists a scalar L such that $\|\mathbf{d}^i\|_2 \leq L$ for any $\mathbf{x} \in \mathbb{R}^N$ and for all $i \in \{1, \ldots, M\}$. Namely, the sub-gradients of all nodes in the network are bounded.*

Then, we recall the following lemma from [19]:

Lemma 2. *Let $0 < \beta < 1$ and let $\{\alpha(k)\}_{k\geq 0}$ be a positive scalar sequence. Assume that $\lim_{k\to+\infty} \alpha(k) = 0$. Then*

$$\lim_{k\to+\infty} \sum_{l=0}^{k} \beta^{k-l}\alpha(l) = 0 \ .$$

In addition, if $\sum_{k=1}^{\infty} \alpha(k) < \infty$, then

$$\sum_{k=1}^{\infty}\sum_{l=0}^{k} \beta^{k-l}\alpha(l) < \infty \ .$$

Using lemma 2 we can prove that

Proposition (Convergence of Agent Estimates)**.** *Under assumption 1 and decreasing step-size rule (see Sec. 5), the estimate of each node converges almost surely to the vector $\mathbf{y}(k)$, i.e.*

$$\lim_{k\to+\infty} \|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 = 0 \quad a.\ s.\ , \text{for all } i \ .$$

Proof. Iterating equation (9), and considering only the $i-$th row of $\mathbf{X}(k)$, we obtain

$$\begin{aligned}\mathbf{x}^i(k) = &\sum_{j=1}^{M} \left[\mathbf{A}_{(1)}^{(k-1)}\right]_{ij} \mathbf{x}^j(1) - \sum_{s=1}^{k-2}\sum_{j=1}^{M} \left[\mathbf{A}_{(s+1)}^{(k-1)}\right]_{ij} \gamma_j(s)\mathbf{d}^j(s) + \\ &- \gamma_i(k-1)\mathbf{d}^i(k-1) \ . \end{aligned} \tag{15}$$

Recalling that $\mathbf{y}(k) \stackrel{\text{def}}{=} 1/M\mathbf{1}^T\mathbf{X}(k)$, we can write

$$\mathbf{y}(k+1) = \mathbf{y}(k) - \frac{1}{M}\sum_{j=1}^{M} \gamma_j(k)\mathbf{d}^j(k) \tag{16}$$

and iteratively

$$\mathbf{y}(k) = \frac{1}{M}\sum_{j=1}^{M} \mathbf{x}^j(1) - \frac{1}{M}\sum_{s=1}^{k-1}\sum_{j=1}^{M} \gamma_j(s)\mathbf{d}^j(s) \ . \tag{17}$$

From (15) and (17) we have

$$
\begin{aligned}
&\|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 = \\
&= \left\| \frac{1}{M} \sum_{j=1}^{M} \mathbf{x}^j(1) - \sum_{j=1}^{M} \left[\mathbf{A}_{(1)}^{(k-1)}\right]_{ij} \mathbf{x}^j(1) + \right. \\
&- \frac{1}{M} \sum_{s=1}^{k-1} \sum_{j=1}^{M} \gamma_j(s) \mathbf{d}^j(s) + \\
&\left. + \sum_{s=1}^{k-2} \sum_{j=1}^{M} \left[\mathbf{A}_{(s+1)}^{(k-1)}\right]_{ij} \gamma_j(s) \mathbf{d}^j(s) + \gamma_i(k-1) \mathbf{d}^i(k-1) \right\|_2 \leq \\
&\leq \sum_{j=1}^{M} \|\mathbf{x}^j(1)\|_2 \left\| \frac{1}{M} - \left[\mathbf{A}_{(1)}^{(k-1)}\right]_{ij} \right\|_2 + \\
&+ \sum_{s=1}^{k-2} \sum_{j=1}^{M} \|\gamma_j(s)\|_2 \left\| \frac{1}{M} - \left[\mathbf{A}_{(s+1)}^{(k-1)}\right]_{ij} \right\|_2 \|\mathbf{d}^j(s)\|_2 + \\
&+ \frac{1}{M} \sum_{j=1}^{M} \|\gamma_j(k-1)\|_2 \|\mathbf{d}^j(k-1)\|_2 + \\
&+ \|\gamma_i(k-1)\|_2 \|\mathbf{d}^i(k-1)\|_2 \leq \\
&\leq \sum_{j=1}^{M} \|\mathbf{x}^j(1)\|_2 b(k-1,1) + L \sum_{j=1}^{M} \sum_{s=1}^{k-2} \gamma_j(s) b(k-1, s+1) + \\
&+ L \left(\frac{1}{M} \sum_{j=1}^{M} \gamma_j(k-1) + \gamma_i(k-1) \right) \ , \qquad (18)
\end{aligned}
$$

where the last inequality follows from the bounded sub-gradient assumption 2 and defining the quantity

$$
b(k,s) \stackrel{\text{def}}{=} \max_{i,j} \left| \left[\mathbf{A}_{(s)}^{(k)}\right]_{ij} - \frac{1}{M} \right| \quad \text{for all } k \geq s \ .
$$

Immediately we note that the last term in the right hand side (rhs) of equation (18) goes to zero as k goes to infinity, since by assumption $\lim_{k\to\infty} \gamma_j(k)$, for all j. From the proof of Proposition 3 we have that for almost all the sequences $\{\mathbf{A}(k)\}_{k\geq 1}$

$$
b(k,s) \leq C\beta^{k-s} \quad \text{for all } k \geq s \ , \qquad (19)
$$

where $C > 0$ and $0 < \beta < 1$ (with C in general depending of the considered sequence). Therefore for almost all the sequences also the first term in the rhs of (18) goes to zero increasing k. Finally, we if (19) holds, Lemma 2 applies and $\lim_{k\to+\infty} \sum_{s=1}^{k-2} \gamma_j(s) b(k-1, s+1) = 0$ for all j. Thus, for almost all the sequences, for all i we have that

$$
0 \leq \lim_{k\to+\infty} \|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 \leq 0 \ ,
$$

proving the desired result. □

F Derivation of Equation (11)

In this Appendix we derive equation (11). To this end, let us consider the following lemma, that is a generalization of Lemma 5 in [3].

Lemma 3 (Basic Iterate Relation)**.** *Let $\mathbf{x}^i(k)$ be generated according to (15) for all $i \in \{1,\dots,M\}$, $k \geq 1$ and $\mathbf{y}(k)$ be generated according to (16) for all $k \geq 1$. Let also $\{\mathbf{g}^i(k)\}_{k\geq 1}$ be a sequence of sub-gradient of $f_i(\cdot)$ computed in $\mathbf{y}(k)$, for all $i \in \{1,\dots,M\}$, then for any $\mathbf{x} \in \mathbb{R}^N$ and $k \geq 1$ we have*

$$\begin{aligned}\|\mathbf{y}(k+1)-\mathbf{x}\|_2^2 \leq \|\mathbf{y}(k)-\mathbf{x}\|_2^2 + \\ + \frac{2}{M}\sum_{j=1}^{M}\gamma_j(k)\left[\left(\|\mathbf{d}^j(k)\|_2 + \|\mathbf{g}^j(k)\|_2\right)\|\mathbf{y}(k)-\mathbf{x}^j(k)\|_2\right] + \\ - \frac{2}{M}\sum_{j=1}^{M}\gamma_j(k)\left[f_j(\mathbf{y}(k)) - f_j(\mathbf{x})\right] + \frac{1}{M^2}\sum_{j=1}^{M}\gamma_j^2(k)\|\mathbf{d}^j(k)\|_2^2 \ .\end{aligned}$$

Proof. It follows straightforwardly from the same rationale of Lemma 5 in [3]. Considering equation (16), we can write, for any $\mathbf{x} \in \mathbb{R}^N$ and all $k \geq 1$

$$\|\mathbf{y}(k+1)-\mathbf{x}\|_2^2 = \left\|\mathbf{y}(k) - \frac{1}{M}\sum_{j=1}^{M}\gamma_j(k)\mathbf{d}^j(k) - \mathbf{x}\right\|_2^2 ,$$

implying that

$$\begin{aligned}\|\mathbf{y}(k+1)-\mathbf{x}\|_2^2 \leq \|\mathbf{y}(k)-\mathbf{x}\| - \frac{2}{M}\sum_{j=1}^{M}\gamma_j(k)\left\{\mathbf{d}^j(k)\right\}^T(\mathbf{y}(k)-\mathbf{x}) + \\ + \frac{1}{M^2}\sum_{j=1}^{M}\gamma_j^2(k)\|\mathbf{d}^j(k)\|_2^2 \ . \qquad (20)\end{aligned}$$

Considering the term $\left\{\mathbf{d}^j(k)\right\}^T(\mathbf{y}(k)-\mathbf{x})$, for any j, we have

$$\begin{aligned}\left\{\mathbf{d}^j(k)\right\}^T(\mathbf{y}(k)-\mathbf{x}) &= \left\{\mathbf{d}^j(k)\right\}^T(\mathbf{y}(k)-\mathbf{x}^j(k)) + \left\{\mathbf{d}^j(k)\right\}^T(\mathbf{x}^j(k)-\mathbf{x}) \geq \\ &\geq -\|\mathbf{d}^j(k)\|_2\|\mathbf{y}(k)-\mathbf{x}^j(k)\|_2 + \left\{\mathbf{d}^j(k)\right\}^T(\mathbf{x}^j(k)-\mathbf{x}) \ .\end{aligned}$$

Since $\mathbf{d}^j(k)$ is a sub-gradient of f_j at $\mathbf{x}^j(k)$, we also have for any j and any $\mathbf{x} \in \mathbb{R}^N$,

$$\left\{\mathbf{d}^j(k)\right\}^T(\mathbf{x}^j(k)-\mathbf{x}) \geq f_j(\mathbf{x}^j(k)) - f_j(\mathbf{x}) \ .$$

Furthermore, by using a sub-gradient $\mathbf{g}_j(k)$ of f_j at $\mathbf{y}(k)$, we obtain for any j and any $\mathbf{x} \in \mathbb{R}^N$,

$$\begin{aligned}f_j(\mathbf{x}^j(k)) - f_j(\mathbf{x}) &= f_j(\mathbf{x}^j(k)) - f_j(\mathbf{y}(k)) + f_j(\mathbf{y}(k)) - f_j(\mathbf{x}) \geq \\ &\geq \left\{\mathbf{g}_j(k)\right\}^T(\mathbf{x}^j(k)-\mathbf{y}(k)) + f_j(\mathbf{y}(k)) - f_j(\mathbf{x}) \geq \\ &\geq -\|\mathbf{g}_j(k)\|_2\|\mathbf{x}^j(k)-\mathbf{y}(k)\|_2 + f_j(\mathbf{y}(k)) - f_j(\mathbf{x}) \ .\end{aligned}$$

By combining the preceding three relations it follows that for any j and any $\mathbf{x} \in \mathbb{R}^N$,

$$\left\{\mathbf{d}^j(k)\right\}^T (\mathbf{y}(k) - \mathbf{x}) \geq -\left(\|\mathbf{d}^j(k)\|_2 + \|\mathbf{g}^j(k)\|_2\right) \|\mathbf{y}(k) - \mathbf{x}^j(k)\|_2 + \\ + f_j(\mathbf{y}(k)) - f_j(\mathbf{x}) \ ,$$

and since $\gamma_j(k) \geq 0$, for all j, it holds also

$$\gamma_j(k) \left\{\mathbf{d}^j(k)\right\}^T (\mathbf{y}(k) - \mathbf{x}) \geq -\gamma_j(k) \left(\|\mathbf{d}^j(k)\|_2 + \|\mathbf{g}^j(k)\|_2\right) \|\mathbf{y}(k) - \mathbf{x}^j(k)\|_2 + \\ + \gamma_j(k) \left[f_j(\mathbf{y}(k)) - f_j(\mathbf{x})\right] \ .$$

Summing this relation over all j, we obtain

$$\sum_{j=1}^{M} \gamma_j(k) \left\{\mathbf{d}^j(k)\right\}^T (\mathbf{y}(k) - \mathbf{x}) \geq \\ \geq -\sum_{j=1}^{M} \gamma_j(k) \left(\|\mathbf{d}^j(k)\|_2 + \|\mathbf{g}^j(k)\|_2\right) \|\mathbf{y}(k) - \mathbf{x}^j(k)\|_2 + \\ + \sum_{j=1}^{M} \gamma_j(k) \left[f_j(\mathbf{y}(k)) - f_j(\mathbf{x})\right] \ .$$

By combining the preceding inequality with equation (20) we finally obtain the desired result, i.e., for all $\mathbf{x} \in \mathbb{R}^N$ and all $k \geq 1$

$$\|\mathbf{y}(k+1) - \mathbf{x}\|_2^2 \leq \|\mathbf{y}(k) - \mathbf{x}\|_2^2 + \\ + \frac{2}{M} \sum_{j=1}^{M} \gamma_j(k) \left[\left(\|\mathbf{d}^j(k)\|_2 + \|\mathbf{g}^j(k)\|_2\right) \|\mathbf{y}(k) - \mathbf{x}^j(k)\|_2\right] + \\ - \frac{2}{M} \sum_{j=1}^{M} \gamma_j(k) \left[f_j(\mathbf{y}(k)) - f_j(\mathbf{x})\right] + \frac{1}{M^2} \sum_{j=1}^{M} \gamma_j^2(k) \|\mathbf{d}^j(k)\|_2^2 \ .$$

□

To carry on our rationale, we need also the following result, that can be proved using Lemma 2 in Appendix E.

Proposition 5. *Under assumption 1 and decreasing step-size rule (see Sec. 5), let $\mathbf{x}^i(k)$ be generated according to (15) for all $i \in \{1, \ldots, M\}$, $k \geq 1$ and $\mathbf{y}(k)$ be generated according to (16) for all $k \geq 1$. Let us also define a sequence $\{\gamma_{\max}(s)\}_{s \geq 1}$ such that $\gamma_{\max}(s) \stackrel{\text{def}}{=} \max_i \gamma_i(s)$ for each $s \geq 1$. Then*

$$\sum_{k=1}^{+\infty} \gamma_{\max}(k) \|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 < \infty \quad \textit{a. s. , for all } i \ .$$

Proof. First of all we note that: 1) since $\{\gamma_i(k)\}_{k \geq 1}$ satisfies $\lim_{k \to \infty} \gamma_i(k) = 0$ for all i, also $\lim_{k \to \infty} \gamma_{\max}(k) = 0$, and 2) since $\{\gamma_i(k)\}_{k \geq 1}$ satisfies $\sum_{k=1}^{\infty} \gamma_i^2(k) < \infty$ for all i in the limited set $\{1, \ldots, M\}$, also $\sum_{k=1}^{\infty} \gamma_{\max}^2(k) < \infty$. In fact, we have that $\sum_{k=1}^{\infty} \gamma_{\max}^2(k) \leq \sum_{k=1}^{\infty} \sum_{i=1}^{M} \gamma_i^2(k) \leq \sum_{i=1}^{M} \sum_{k=1}^{\infty} \gamma_i^2(k) < \infty$.

Now, from equation (18) in Appendix E and using $\gamma_i(s) \leq \gamma_{\max}(s)$ for all i, we have that

$$\|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 \leq \sum_{j=1}^{M} \|\mathbf{x}^j(1)\|_2 b(k-1,1) + \\ + LM \sum_{s=1}^{k-2} \gamma_{\max}(s) b(k-1,s+1) + 2L\gamma_{\max}(k-1) \ ,$$

that, with $C_1 \stackrel{\text{def}}{=} \max\left\{\sum_{j=1}^{M} \|\mathbf{x}^j(1)\|_2, ML\right\}$ and $\gamma_{\max}(0) \stackrel{\text{def}}{=} 1$, can be rewritten as

$$\|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 \leq C_1 \sum_{s=0}^{k-2} \gamma_{\max}(s) b(k-1,s+1) + \\ + 2L\gamma_{\max}(k-1) \ ,$$

multiply at both sides for $\gamma_{\max}(k)$ we have

$$\gamma_{\max}(k)\|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 \leq C_1 \sum_{s=0}^{k-2} \gamma_{\max}(k)\gamma_{\max}(s) b(k-1,s+1) + \\ + 2L\gamma_{\max}(k)\gamma_{\max}(k-1) \ .$$

Recalling Proposition 3 we have that for almost all the sequences $\{\mathbf{A}(k)\}_{k\geq 1}$, $b(k,s) \leq C_2\beta^{k-s}$ for all $k \geq s$, where $C_2 > 0$ and $0 < \beta < 1$ (with C_2 in general depending of the considered sequence). Calling C the product C_1C_2, we have for almost all the sequences

$$\gamma_{\max}(k)\|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 \leq C \sum_{s=0}^{k-2} \gamma_{\max}(k)\gamma_{\max}(s)\beta^{k-s-2} + \\ + 2L\gamma_{\max}(k)\gamma_{\max}(k-1) \ .$$

Noting that $\gamma_{\max}(k)\gamma_{\max}(s) \leq \gamma^2_{\max}(k) + \gamma_{\max}(s)^2$ and $2\gamma_{\max}(k)\gamma_{\max}(k-1) \leq \gamma^2_{\max}(k) + \gamma^2_{\max}(k-1)$, we obtain for almost all the sequences

$$\gamma_{\max}(k)\|\mathbf{y}(k) - \mathbf{x}^i(k)\|_2 \leq C\gamma^2_{\max}(k) \sum_{s=0}^{k-2} \beta^{k-s-2} + \\ + C \sum_{s=0}^{k-2} \gamma^2_{\max}(s)\beta^{k-s-2} + L\gamma^2_{\max}(k) + L\gamma^2_{\max}(k-1) \leq \\ \leq \frac{C\gamma_{\max}(k)^2}{1-\beta} + C \sum_{s=0}^{k-2} \gamma^2_{\max}(s)\beta^{k-s-2} + L\gamma^2_{\max}(k) + \\ + L\gamma^2_{\max}(k-1) \ ,$$

and summing this last inequality over all the k

$$\sum_{k=1}^{+\infty}\gamma_{\max}(k)\|\mathbf{y}(k)-\mathbf{x}^i(k)\|_2 \leq \frac{C}{1-\beta}\sum_{k=1}^{+\infty}\gamma_{\max}^2(k)+$$
$$C\sum_{k=1}^{+\infty}\sum_{s=0}^{k-2}\gamma_{\max}^2(s)\beta^{k-s-2}+L\sum_{k=1}^{+\infty}\gamma_{\max}^2(k)+$$
$$+L\sum_{k=1}^{+\infty}\gamma_{\max}^2(k-1) \quad \text{a. s. .} \tag{21}$$

The desired result follows straightforwardly since: 1) the first, third and fourth term at the right hand side of equation (21) are summable as stated at the beginning of the proof, and 2) the second term of (21) is summable because lemma 2 in Appendix E applies. □

Now we have all the tools to derive equation (11). Applying iteratively lemma 3 we have

$$\|\mathbf{y}(k+1)-\mathbf{x}\|_2^2 \leq \|\mathbf{y}(1)-\mathbf{x}\|_2^2+$$
$$+\frac{2}{M}\sum_{s=1}^{k}\sum_{j=1}^{M}\gamma_j(s)\left[\left(\|\mathbf{d}^j(s)\|_2+\|\mathbf{g}^j(s)\|_2\right)\|\mathbf{y}(s)-\mathbf{x}^j(s)\|_2\right]+$$
$$-\frac{2}{M}\sum_{s=1}^{k}\sum_{j=1}^{M}\gamma_j(s)\left[f_j(\mathbf{y}(s))-f_j(\mathbf{x})\right]+\frac{1}{M^2}\sum_{s=1}^{k}\sum_{j=1}^{M}\gamma_j^2(s)\|\mathbf{d}^j(s)\|_2^2 .$$

and then we can write

$$\sum_{s=1}^{k}\sum_{j=1}^{M}\gamma_j(s)\left[f_j(\mathbf{y}(s))-f_j(\mathbf{x})\right] \leq \frac{M}{2}\|\mathbf{y}(1)-\mathbf{x}\|_2^2+$$
$$+\sum_{s=1}^{k}\sum_{j=1}^{M}\gamma_j(s)\left[\left(\|\mathbf{d}^j(s)\|_2+\|\mathbf{g}^j(s)\|_2\right)\|\mathbf{y}(s)-\mathbf{x}^j(s)\|_2\right]+$$
$$+\frac{1}{2M}\sum_{s=1}^{k}\sum_{j=1}^{M}\gamma_j^2(s)\|\mathbf{d}^j(s)\|_2^2 \leq$$
$$\leq \frac{M}{2}\|\mathbf{y}(1)-\mathbf{x}\|_2^2+2L\sum_{s=1}^{k}\sum_{j=1}^{M}\gamma_j(s)\|\mathbf{y}(s)-\mathbf{x}^j(s)\|_2+$$
$$+\frac{L^2}{2M}\sum_{s=1}^{k}\sum_{j=1}^{M}\gamma_j^2(s) ,$$

where the last inequality follows from the bounded sub-gradient assumption 2. Using the sequence $\{\gamma_{\max}(s)\}_{s\geq 1}$ defined in the Proposition 5 of above and

rearranging some terms, we obtain

$$\sum_{s=1}^{k}\sum_{j=1}^{M}\gamma_j(s)\left[f_j(\mathbf{y}(s))-f_j(\mathbf{x})\right]\leq$$
$$\leq\frac{M}{2}\|\mathbf{y}(1)-\mathbf{x}\|_2^2+2L\sum_{j=1}^{M}\sum_{s=1}^{k}\gamma_{\max}(s)\|\mathbf{y}(s)-\mathbf{x}^j(s)\|_2+$$
$$+\frac{L^2}{2M}\sum_{j=1}^{M}\sum_{s=1}^{k}\gamma_{\max}^2(s)\ .$$

For k that goes to infinity, $\sum_{s=1}^{k}\gamma_{\max}^2(s)<\infty$ as note in the proof of Proposition 5, whilst $\sum_{s=1}^{k}\gamma_{\max}(s)\|\mathbf{y}(s)-\mathbf{x}^j(s)\|_2<\infty$ for the Proposition 5 itself. $\frac{M}{2}\|\mathbf{y}(1)-\mathbf{x}\|_2^2$ is a constant term, therefore equation (11) follows.

G Derivation of the Utility Function in Eq. (13)

In this Appendix we briefly review the mathematical steps of the analysis carried on in [7], under the assumptions done in Section 6 and with the formalism introduced therein. The overall objective of this Appendix is to formally derive equation (13).

For a given time $\bar{t}$ and node $i \in \mathcal{N}$ we define the process $B_i(\bar{t}, t) \subseteq \mathcal{N}$ as the set containing all the nodes j such that, if a message is given to them at time $\bar{t} - t$, it can reach user i in two hops by time $\bar{t}$. For each node pair (i, j), $j \neq i, j, i \in \mathcal{N}$, we define also $s_{ij}(\bar{t})$ as

$$s_{ij}(\bar{t}) \stackrel{\text{def}}{=} \inf_{t \geq 0} \{t : j \in B_i(\bar{t}, t)\} \ .$$

$\bar{t} - s_{ij}(\bar{t})$ indicates the minimum amount of time required to transfer information from node j to node i at time $\bar{t}$ through file sharing. If we assume that the intermeeting process time between the node pair (i, j) is exponentially distributed with parameter λ_{ij} (i.e., Poisson meeting process) we have that in the case of two-hops protocol also $s_{ij}(\bar{t})$ is exponentially distributed with parameter λ_{ij}.

Let then $Y_i^{SP}(\bar{t})$ be, at time $\bar{t}$, the elapsed time since user i downloaded content directly from the SP, i.e., $Y_i^{SP}(\bar{t}) = \bar{t} - t_i^{SP}(\bar{t})$. Note that the random variable $Y_i^{SP}(\bar{t})$ for all $i \in \mathcal{N}$ is exponentially distributed with parameter x_i, this because the SP transfers updates directly to node i with rate x_i and in stationary conditions the forward process and the backward one have the same statistic, see e.g., [21].

Lemma 1 of [7] states that

$$Y_i(\bar{t}) = \min_{j \in \mathcal{N}} \left\{ s_{ij}(\bar{t}) + Y_j^{SP}(\bar{t} - s_{ij}(\bar{t})) \right\} \ , \tag{22}$$

therefore, in our case, $Y_i(\bar{t})$ is the minimum over M independent random variables. One of this random variables is exponentially distributed with parameter x_i (i.e., $s_{ij}(\bar{t}) + Y_j^{SP}(\bar{t} - s_{ij}(\bar{t}))$ with $j = i$) and takes in account the directed updates of i from the SP. The remaining $M - 1$ random variables are sum of two independent exponential random variables: one distributed with parameter λ_{ij} and the second with parameter x_j. Each of these $M - 1$ variables models the update of i in two hops from the PS through a given relay node $j \neq i$. For the independence of s_{ij} and Y_j^{SP}, and assuming in general $\lambda_{ij} \neq x_j$, we have that

$$p_{s_{ij} + Y_j^{SP}}(y) = \lambda_{ij} x_j \left[\frac{e^{-\lambda_{ij} y} - e^{-x_j y}}{x_j - \lambda_{ij}} \right] \quad y > 0 \ ,$$

whence

$$P\left[s_{ij} + Y_j^{SP} > y\right] = \frac{x_j e^{-\lambda_{ij} y} - \lambda_{ij} e^{-x_j y}}{x_j - \lambda_{ij}} \ ,$$

therefore

$$P\left[Y_i > y\right] = \left[\prod_{j \in \mathcal{N}_i} \frac{x_j e^{-\lambda_{ij} y} - \lambda_{ij} e^{-x_j y}}{x_j - \lambda_{ij}} \right] e^{-x_i y} \ ,$$

where $\mathcal{N}_i \stackrel{\text{def}}{=} \{j : j \neq i, \lambda_{ij} > 0\}$. In the case of utility $u(Y_i) = \chi\{Y_i \leq \tau\}$ we have that

$$\begin{aligned} f_i(\mathbf{x}) &= E_{\mathbf{x}}[u_i(Y_i)] = P[Y_i \leq \tau] = \\ &= 1 - \left[\prod_{j \in \mathcal{N}_i} \frac{x_j e^{-\lambda_{ij}\tau} - \lambda_{ij} e^{-x_j \tau}}{x_j - \lambda_{ij}}\right] e^{-x_i \tau} , \end{aligned}$$

thus

$$\begin{aligned} f(\mathbf{x}) &= \sum_{i=1}^{M} f_i(\mathbf{x}) = \\ &= M - \sum_{i=1}^{M} \left[\prod_{j \in \mathcal{N}_i} \frac{x_j e^{-\lambda_{ij}\tau} - \lambda_{ij} e^{-x_j \tau}}{x_j - \lambda_{ij}}\right] e^{-x_i \tau} . \end{aligned}$$

From equation (13) is straightforward to compute the components of the gradient vector we need to implement the distributed sub-gradient method, for all i:

$$\frac{\partial f_i(\mathbf{x})}{\partial x_i} = \tau \left[\prod_{j \in \mathcal{N}_i} \frac{x_j e^{-\lambda_{ij}\tau} - \lambda_{ij} e^{-x_j \tau}}{x_j - \lambda_{ij}}\right] e^{-x_i \tau} ,$$

and, for $z \neq i$

$$\begin{aligned} \frac{\partial f_i(\mathbf{x})}{\partial x_z} &= \left[\prod_{\substack{j \in \mathcal{N}_i \\ j \neq z}} \frac{x_j e^{-\lambda_{ij}\tau} - \lambda_{ij} e^{-x_j \tau}}{x_j - \lambda_{ij}}\right] e^{-x_i \tau} \cdot \\ &\quad \cdot \frac{\lambda_{iz}\left\{e^{-\lambda_{iz}\tau} + [\tau(\lambda_{iz} - x_z) - 1] e^{-x_z \tau}\right\}}{(x_z - \lambda_{iz})^2} . \end{aligned}$$

References

[1] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Gossip algorithms: design, analysis and applications," in *INFOCOM 2005. 24th Annual Joint Conf. of the IEEE Comp. and Comm. Soc. Proceedings IEEE*, Miami, Florida, USA, 2005.

[2] S.M. Hedetniemi and S.T. Hedetniemi and A.L. Liestman, "A Survey of Gossiping and Broadcasting in Communication Networks," *Networks*, vol. 18, no. 4, pp. 319–349, 1988.

[3] A. Nedic and A. Ozdaglar, "Distributed Subgradient Methods for Multi-Agent Optimization," *Automatic Control, IEEE Transactions on*, vol. 54, no. 1, pp. 48–61, Jan. 2009.

[4] I. Lobel and A. Ozdaglar, "Distributed Subgradient Methods for Convex Optimization over Random Networks," *Automatic Control, IEEE Transactions on - Accepted for publication*, May 2010.

[5] R. Wei, R. Beard, and E. Atkins, "A survey of Consensus Problems in Multi-agent Coordination," *American Control Conference, 2005. Proceedings of the 2005*, vol. 3, pp. 1859–1864, Jun. 2005.

[6] P. Denantes, F. Bénézit, P. Thiran, and M. Vetterli, "Which Distributed Averaging Algorithm Should I Choose for my Sensor Network?" in *Proc. 27th IEEE Conf. Computer Communications and Networks*, 2008.

[7] S. Ioannidis, A. Chaintreau, and L. Massoulie, "Optimal and Scalable Distribution of Content Updates over a Mobile Social Network," in *IEEE INFOCOM 2009*, Rio de Janeiro, Brazil, Apr. 2009, pp. 1422–1430.

[8] S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, K. Scott, and H. Weiss, "Delay-tolerant Networking: an Approach to Interplanetary Internet," *Communications Magazine, IEEE*, vol. 41, no. 6, pp. 128–136, Jun. 2003.

[9] L. Pelusi, A. Passarella, and M. Conti, "Opportunistic Networking: Data Forwarding in Disconnected Mobile Ad Hoc Networks," *Communications Magazine, IEEE*, vol. 44, no. 11, pp. 134–141, Nov. 2006.

[10] T. Spyropoulos, K. Psounis, and C. Raghavendra, "Efficient Routing in Intermittently Connected Mobile Networks: The Multiple-Copy Case," *Networking, IEEE/ACM Transactions on*, vol. 16, no. 1, pp. 77–90, Feb. 2008.

[11] G. Neglia, G. Reina, and S. Alouf, "Distributed Gradient Optimization for Epidemic Routing: A Preliminary Evaluation," in *Wireless Days (WD), 2009 2nd IFIP*, Paris, FR, Dec. 2009.

[12] C. Bettstetter, "Mobility Modeling in Wireless Networks: Categorization, Smooth Movement, and Border Effects," *ACM Mobile Comp. Comm. Rev.*, vol. 5, no. 3, pp. 55–67, 2001.

[13] A. N. Shiryaev, *Probability (2nd ed.)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 1995, translator-Boas, R. P.

[14] R.C. Bradley, "Basic Properties of Strong Mixing Conditions. A Survey and Some Open Questions," *Probability Surveys*, 2005.

[15] A. Tahbaz-Salehi and A. Jadbabaie, "Consensus Over Ergodic Stationary Graph Processes," *Automatic Control, IEEE Transactions on*, vol. 55, no. 1, pp. 225–230, Jan. 2010.

[16] E. Seneta, *Non-negative Matrices and Markov Chains*. Springer, New York 2^{rd} edition, 1981.

[17] J.C. Oxtoby, "Ergodic sets," *Bulletin of the American Mathematical Society*, vol. 58, pp. 116–136, 1952.

[18] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[19] A. Nedic, A. Ozdaglar, and P. Parrilo, "Constrained Consensus and Optimization in Multi-Agent Networks," *Automatic Control, IEEE Transactions on*, Feb. 2010, accepted for publication.

[20] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized Gossip Algorithms," *Information Theory, IEEE Transactions on*, vol. 52, no. 6, pp. 2508–2530, Jun. 2006.

[21] S.M. Ross, *Introduction to Probability Models*. Elsevier, Academic Press, 2007.

Contents

Centre de recherche INRIA Sophia Antipolis – Méditerranée
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Saclay – Île-de-France : Parc Orsay Université - ZAC des Vignes : 4, rue Jacques Monod - 91893 Orsay Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399